

ANÁLISIS Y DISEÑO DE INTERFAZ PARA CONEXIÓN DE PLC BASADO EN
ARDUINO CON PLC ALLEN BRADLEY A TRAVÉS DE ETHERNET.

Jose Mario Valencia Henao.

Universidad Tecnológica de Pereira

Facultad de Ingenierías

Ingeniería de Sistemas y computación

Pereira, Risaralda.

2018

ANÁLISIS Y DISEÑO DE INTERFAZ PARA CONEXIÓN DE PLC BASADO EN
ARDUINO CON PLC ALLEN BRADLEY A TRAVÉS DE ETHERNET.

Jose Mario Valencia Henao.

Asesor:

PhD. Guillermo Roberto Solarte Martinez

Doctor.

Trabajo de investigación formativa como requisito para optar al título de:

Ingeniero de Sistemas y Computación.

Universidad Tecnológica de Pereira

Facultad de Ingenierías

Ingeniería de Sistemas y computación

Pereira, Risaralda.

2018

Agradecimientos especiales

Antes de comenzar con este proyecto quiero agradecer primero que todo a la gran Universidad Tecnológica de Pereira, y a todos sus colaboradores, juntos me ha brindado todas las herramientas y conocimientos necesarios para la Ingeniería de sistemas, pero sobre todo para la vida, por hacer de mi no solo un profesional, sino una persona consciente de los problemas sociales y culturales que afectan la región cafetera y de los utensilios necesarios para analizarlos y atacarlos.

Agradecimiento inmenso a mi director de proyecto, PhD. Guillermo Solarte, un gran profesional y una gran persona que he tenido la fortuna y la oportunidad de conocer desde mis inicios en la carrera, el cual ha generado en mí los deseos y los conocimientos necesarios para llevar a cabalidad la finalización de este proyecto.

A mi padre, el Ingeniero Mario Valencia, orgulloso egresado de la facultad de ingenierías en el programa Ingeniería Eléctrica de la Universidad Tecnológica de Pereira, cuya guía, apoyo y consejo ha estado para mí en los momentos más difíciles de mi vida y mi carrera, fue él quien me apoyó desde el inicio de este proyecto que se llamó Ingeniero de Sistemas, y sus conocimientos amplios en ciencias básicas, matemáticas y Automatización Industrial, fueron las bases iniciales de este proyecto de Investigación, gracias por ser siempre mi héroe.

A mi madre, Luz Stella Henao, un apoyo incondicional en todas las etapas de mi vida, el ángel que Dios puso para alumbrar mi camino y jamás permitir que desistiera de mis deseos de ser profesional. Gracias por ser la fiel representación de Dios en la tierra.

A mi familia, mis hermanos y mis amigos que han sabido apoyarme y entenderme en las buenas y malas, personas de gran valor que con su gran amor le han dado gran sentido a mi vida y motivación para terminar este proyecto.

Tabla de contenido

- I. Resumen**
- II. Introducción.**
- III. Planteamiento del problema.**
- IV. Formulación del problema.**
- V. Objetivos**
 - 1. Objetivo general
 - 2. Objetivos específicos del proyecto
- VI. Justificación del proyecto.**
- VII. Marco referencial y teórico**
 - 1. Glosario de términos
- VIII. Diseño metodológico**
 - A. Estrategia metodológica
 - B. Cronograma de tiempos y tareas
 - C. Presupuesto de costos del proyecto
 - D. Participantes del proceso
 - E. Procedimiento
 - 1. Etapa 1: Investigación y captación de información.***
 - a) Actividad 1, Investigar documentación relacionada a la creación de una API, SOA e interfaz web
 - b) Actividad 2, Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen Bradley y de un PLC basado en arduino.
 - c) Actividad 3, Analizar cómo cada dispositivo ejerce control sobre el sistema

2. Etapa 2: Diseño y divulgación.

a) Actividad 1, Diseñar una interfaz que permita administrar PLC Allen Bradley desde un PLC basado en arduino

(1) Tarea 1, Levantamiento de requerimientos

(2) Tarea 2, Elaboración de casos de uso

(3) Tarea 3, Desarrollo de especificación

IX. Anexos

A. Historia de los PLC

B. Estado del arte

C. Ciclo de programa de un PLC

1. Kunbus

2. Industrial Shields

D. Configurar un tópico OPC/DDE en RsLinx para obtener datos de un controlador Logix 500 y agregarlo a una hoja de cálculo de excel

X. Resultados

XI. Discusión

XII. Citas y Bibliografía

Listas especiales

Tablas

1. Tabla 1, Comparación de características especiales de los 3 tipos de módulos PLCs Core de la empresa Kunbus
2. Tabla 2, Comparación de las características principales de los módulos de E / S disponibles de los PLCs módulo base de la empresa Kunbus
3. Tabla 3, Gateways de la empresa Kunbus para conexión a otras redes industriales bajo los más conocidos protocolos
4. Tabla 4, Comparación de las diferentes tecnologías de controladores actualmente disponibles
5. Tabla 5 comparativa de los dos tipos de PLC all in de Industrial Shields, su única diferencia es la salida
6. Tabla 6, Comparación de dispositivos PLC de Industrial Shields con conexión a Ethernet que tienen la capacidad de comunicarse a través de Ethernet
7. Tabla 7, Tabla comparativa de los cuatro tipos de dispositivos de esta rama HMI o Panel PC de Industrial Shields
8. Tabla 8, Documentación de los actores que interactúan en los casos de uso
9. Tabla 9, Lista de drivers arquitectónicos

Figuras

1. Figura 1, Gráfico de la pirámide de automatización básica de 4 niveles.
2. Figura 2, Cronograma presentado en la ficha técnica de radicación del proyecto
3. Figura 3, Integración de sistemas en un entorno heterogéneo se torna sencillo a través de OPC
4. Figura 4, Imagen de un PLC Allen Bradley Micrologix 1100 Serie B
5. Figura 5, Imagen de M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS
6. Figura 6, Dibujo de visión gráfica del sistema controlado por PLC AB.
7. Figura 7, Ejemplo de programación de dos salidas y su activación
8. Figura 8, plano de conectividad de un sistema ejemplo con varios Plc, una pantalla y un router
9. Figura 9, ilustración de los lenguajes de programación usados en una arquitectura con dispositivos de industrial shields
10. Figura 10, Bosquejo del sistema en base a un gráfico de arquitectura OPC
11. Figura 11, Diagrama UML de los casos de uso
12. Figura 12, Esquema gráfico acercado de los elementos de desarrollo
13. Figura A1, Seleccionador de ruta Controller Tag.
14. Figura A2, array DINT contenedor de datos a exportar
15. Figura A3, Variable temperatura que toma datos cambiantes provenientes de un sensor de calor
16. Figura A4, seleccionador del controlador número 3, el cual contiene los datos a exportar

17. Figura A5, selección de opción para configuración del tópico DDE/OPC
18. Figura A6, Configuración de primer tópico llamado Production lado izquierdo de la figura, lado derecho controlador al SCADA que apunta.
19. Figura A7, copia del link DDE/OPC en RSLinx.
20. Figura A8 , selección de la variable “temperature” para inicializar la conexión
21. Figura A9, Ventana emergente que se despliega luego de hacer click a la opción copiado especial.
22. Figura A10, Secuencia izquierda a derecha de capturas de pantalla en la celda E5 hecha con 3 segundos de retraso entre sí
23. Figura A 11, Selección de los datos del array DINT_Array[]
24. Figura A12, Captura de pantalla después de pegado especial de datos del array en bloque 1 x 10

Resumen

En las próximas páginas de este proyecto de investigación, el lector puede encontrar el planteamiento para una interfaz que permita la conexión entre equipos PLC de diferente fabricante y filosofía de lucro de software, por un lado tenemos al gigante de la automatización industrial Allen Bradley, que tiene bastante presencia en nuestra región, y cuenta con una suite de software privativo que incrementa su costo proporcionalmente con su funcionalidad, y por otro lado Industrial Shields, una empresa que ha surgido en la búsqueda por equipos de bajo costo, cobijada por el apogeo y el boom de la tecnología basada en código abierto, la cual ofrece equipos con características de soporte industriales, programables en plataformas Open Source; todo esto con el fin de aprovechar las infraestructuras actualmente instaladas basadas en el software con propietario para la captación de datos físicos y abrir las barreras de limitaciones de software para que los datos de producción puedan ser exportados y visualizados a gusto y nivel de los ejecutivos, sin pensar en adquirir software en el rango de las decenas de millones de pesos sino invirtiendo en conocimiento aprovechando las puertas que se abren al programar en plataformas gratuitas y abiertas a la comunidad de programadores multidisciplinariamente.

Palabras clave: *Automatización Industrial, Open Source, PLC, Arduino, SCADA, Interfaz OPC*

Introducción

En el mundo de la industria, es común encontrar dentro de las grandes fábricas y empresas, que muchos de sus procesos, desde los operacionales hasta los de producción, han sido tecnificados y estandarizados utilizando herramientas en el mercado como Computadores, Sistemas embebidos, lógica cableada y Controladores Lógicos Programables. Estos últimos, mundialmente conocidos como PLC, se han vuelto el pilar de la industrialización de procesos de producción, puesto que son computadoras sin periféricos E/S, optimizadas con mucha robustez para la transmisión y recepción de datos en ambientes hostiles y de mucho ruido en términos de señales, que brindan a quien lo usa, la seguridad y confiabilidad de los datos procesados que viajan desde la máquina a la estación de control, y viceversa.

La fabricación y programación de controladores lógicos programables, o PLCs, ha sido comandada por los grandes fabricantes internacionales, Siemens en Europa, Allen Bradley en estados unidos, Omron en Japón, Delta en China, entre otros, desarrollan junto con sus equipos, software que se encarga de gestionar los datos y la comunicación entre estos, por consiguiente cada fabricante desarrolla software que es ajustado a las características de cada dispositivo, sin embargo, un sistema basado en PLCs, es un sistema cerrado, hermético, que poco contacto tiene con el mundo exterior, solo lo que el fabricante y las normativas lo permitan, en conclusión, un desarrollo con un PLC es muy ajustado en términos de escalabilidad y alcance a las limitantes de cada fabricante, comparado a los grandes costos que acarrea.

Planteamiento del problema

Hoy en día el software libre ha llegado a lugares impensables, y, es en este punto en donde un fabricante de PLCs se propuso a instalar placas arduino en estos controladores lógicos, desde el 2006, abriendo paso a un nuevo mundo de automatizaciones industriales basadas en internet de las cosas, este es un gran aliciente para los profesionales en sistemas

que tenemos afinidad o interés con el mundo de la automatización, no sólo para desarrollar nuevos sistemas automatizados bajo la filosofía del software libre, sino también para migrar los sistemas actualmente instalados y aumentar las funcionalidades ya obtenidas, con las ventajas de la programación en plataformas abiertas.

Por años, el sector industrial ha automatizado operaciones usando dispositivos PLC, en latinoamérica el fabricante líder ha sido Allen bradley, con tecnología de punta, y equipos compactos de mediana gama, como el Micrologix 1100, pero costoso, alrededor de 4 millones por equipo, y software como el RS Logix 500 que comienza desde los 15 millones de pesos en adelante por estación, es decir, de manera local , lo que hace que la automatización de cualquier proceso signifique una mediana o gran inversión, y pensar en un sistema distribuido con varias estaciones hace que sea una inversión monumental.

Formulación del problema

En nuestra región, debido al nublado concepto cultural que se ve en la inversión en tecnología, puesto que el retorno a capital en algunos casos puede tomar décadas, son contadas las empresas que cuentan con sistemas de automatización industrial estandarizados a nivel mundial, importados e instalados por empresas como G&L Ingenieros, que llevan gran trayectoria en la instalación y acoplamiento de sistemas automatizados, pero que por consiguiente suponen grandes inversiones en capital para quien los contrate, no sólo para adquirir estos costosos equipos, también para suplir los costos de mano de obra tecnificada.

El inconveniente principal a la hora de pensar en migrar sistemas completos que ya cumplen de manera óptima el objetivo que fueron diseñados, a software libre, es el costo, puesto que los directivos ya han aprobado los presupuestos, y las fábricas poseen equipos e infraestructura de alguno(s) de los fabricantes ya mencionados anteriormente y que llamaremos “cerrados”, y por esto, plantear un cambio en el cerebro (PLC) del proyecto, significaría idear un proyecto nuevo con costos que ya fueron asumidos y ejecutados, para solucionar un problema que ya fue corregido; ningún directivo en sus 5 sentidos estaría dispuesto a re - asumir costos innecesarios.

Objetivos

Objetivo general del proyecto

Analizar y diseñar una interfaz de conexión entre un(os) dispositivo(s) PLC allen bradley y un PLC basado en arduino de la empresa industrial shields.

Objetivos específicos:

1. Investigar documentación relacionada a la creación de una API, SOA e interfaz web.
2. Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen bradley y un PLC basado en arduino.
3. Analizar cómo cada dispositivo PLC ejerce control sobre el sistema.
4. Diseñar una interfaz que permita administrar PLCs Allen bradley desde un PLC basado en arduino.
5. Comunicación y divulgación de los resultados obtenidos de la investigación.

Justificación del proyecto

Por lo anteriormente visto en este capítulo, para que las empresas innoven sin pensar en rehacer, bajo la filosofía de Open Source, que significa una alta reducción de costos y mano de obra, es que surge la necesidad de mover progresivamente estos sistemas cerrados, a sistemas basados en software libre, sin afectar la productividad de la empresa. Todo esto usando la infraestructura física de control que se encuentre instalada en ese momento, para crear una interfaz que permita conectar el PLC del fabricante cerrado Allen Bradley, y este a su vez actúe como esclavo del PLC basado en placas arduino, cuyo soporte industrial y programación de software es gratuito. El límite de los proyectos de automatización industrial se quebrará, al entrar a un universo recientemente creado y poco explorado en nuestra región, automatización de procesos industriales usando equipamiento programable en Arduino, expandiendo los posibles usos y aplicaciones de los costosos equipos que se hayan instalado.

De esta manera no hay que pensar en replantear un proyecto que ya se encuentra ejecutado, por el contrario se estaría ofreciendo una optimización del sistema, con el objetivo principal de aprovechar las ventajas del software libre, como lo son sistemas en la nube y su estrecha relación con el Internet de las cosas en sistemas embebidos.

Marco referencial y teórico

Glosario de términos

PLC. Según Abril Erendira Murillo, (2013), en el siguiente párrafo para el portal CTIN México:

Un PLC es un Controlador Lógico Programable (Programmable Logic Controller), en sí es un sistema de control. Los PLC's son dispositivos electrónicos o computadoras digitales de tipo industrial que permiten la automatización, especialmente de procesos de la industria, debido a que controlan tiempos de ejecución y regulan secuencias de acciones.

De acuerdo con la definición de la NEMA (National Electrical Manufacturers Association) un PLC es: “Un aparato electrónico operado digitalmente, que usa una memoria programable para el almacenamiento interno de instrucciones para implementar funciones específicas, tales como lógica, secuenciación, registro y control de tiempos, conteo y operaciones aritméticas para controlar, a través de módulos de entrada/salida digitales (ON/OFF) o analógicos (1-5 VDC, 4-20 mA, etc.), varios tipos de máquinas o procesos”.

Para efectos prácticos, se resume que un PLC es una computadora robusta y confiable, cuyos periféricos de entrada son sensores y de salida son actuadores, diseñada para estar en ambientes hostiles en términos de señales, y transmitir datos con un nivel de confiabilidad muy alto, pueden ser programadas en el sitio de trabajo o remotamente según sea su arquitectura, se pueden encontrar de muchas marcas, tipos y especificaciones, y por lo general el software que utiliza debe ser adquirido a la misma empresa que lo fabrica.

SCADA. *(Supervisory Control And Data Acquisition traducido a Supervisión, Control y Adquisición de Datos) es un estilo de software para ordenadores que permite controlar y supervisar procesos industriales a distancia. Facilita retroalimentación en tiempo real con los dispositivos de campo (sensores y actuadores), y controla el proceso automáticamente. Provee de toda la información que se genera en el proceso productivo (supervisión, control calidad, control de producción, almacenamiento de datos, etc.) y permite su gestión e intervención.*

Automatización industrial. La automatización industrial, que viene del griego antiguo auto, ‘guiado por uno mismo’ es el uso de sistemas o elementos computarizados y electromecánicos para fines industriales. va mucho más allá que un mero sistema de control, la instrumentación industrial abarca los sensores, los transmisores de campo, los sistemas de control y supervisión, los sistemas de transmisión y recolección de datos y las aplicaciones de software en tiempo real para supervisar y controlar las operaciones de plantas o procesos industriales.

Interfaz. Para la definición de una interface, el portal sistemas.com (Anonimo) usa las siguientes palabras: “Definimos genéricamente en el mundo de la electrónica a la Interfaz como todo puerto que nos permite enviar y recibir señales desde un componente a otro, teniendo entonces distintas formas de realizar este envío dispuestas por las Especificaciones Técnicas de cada equipo, o bien mediante el establecimiento de distintos estándares que permiten la comunicación.

En este caso particular, la interfaz a desarrollar es una interfaz de software, en la que se comunicará el dispositivo de la empresa Allen Bradley a través del software de Rockwell automation, con un dispositivo basado en placas Arduino de la empresa Industrial Shields a través de Arduino IDE, teniendo cada dispositivo sus propios protocolos de comunicación, en el primer caso protocolos privados, en el segundo caso, basado en software libre, y utilizando las herramientas que han sido dispuestas por los fabricantes de cada equipo.

Arduino. Según la página oficial, (Arduino.cl) se define Arduino como una plataforma de código abierto para prototipado electrónico, basado en hardware y software que son muy flexibles y adaptables a cualquier proyecto.

Debido a su microcontrolador, programable en “Arduino Programming Language” (basado en Wiring) y el “Arduino Development Environment” (basado en Processing), es capaz de sentir el mundo a través de la recepción de entradas conectadas a sensores, y es capaz de actuar sobre él usando luces, motores y otros artefactos, puede ser un software autónomo o estar en comunicación con otro software en ejecución.

La placa física puede ser ensamblada a mano o pedirla preensamblada, por un rango de precios que oscila entre veinticinco mil pesos (\$25.000) y sesenta mil pesos (\$60.000), puede ser ampliada su funcionalidad a través de “shields” que son placas de expansión con funcionalidades como conexión a Ethernet o Bluetooth, el software puede ser descargado gratuitamente y los archivos CAD de referencia están libres bajo licenciamiento Open Source, por lo que pueden ser adaptados a cualquier necesidad.

Allen Bradley. Nombre de la marca de la línea de manufacturación de equipos de automatización industrial, que pertenece al gigante de la industria Rockwell Automation, la cual desarrolla el software para la programación de los equipos, AB. (Allen Bradley por sus siglas) es una compañía que en 2013 tuvo ganancia por aproximadamente \$6.4 billones de Dólares americanos, se dedica a producir PLCs, interfaces hombre máquina (HMI), sensores, componentes de seguridad para los sistemas, software especializado, diversos físicos y de software, contactores, motores, centros de control de equipos y muchos otros.

La empresa Rockwell Automation, propietaria de Allen Bradley, también ofrece servicios que incluyen asesoría y consultoría de proyectos. Las oficinas principales de Rockwell Automation se encuentran en Milwaukee, Wisconsin.

HMI. Human - Machine Interface, Interfaz hombre - máquina, son las siglas que describen todo medio por el que un usuario puede comunicarse con una máquina, equipo, dispositivo, en este caso PLC, y abarca todos los puntos de contacto entre el operario y el equipo. Pueden ser de hardware, como por ejemplo un gateway o una pantalla táctil o de software, como una línea de comandos (shell) dependiendo de las necesidades que tenga el usuario.

Gateway. Una pasarela o puerta de enlace es el dispositivo que actúa como interfaz de conexión física entre aparatos o dispositivos, y también posibilita compartir recursos entre dos o más computadoras.

El propósito principal de estos dispositivos es traducir transferencias de datos entre redes que tienen arquitecturas diferentes, por ejemplo, entre una red Ethernet y una red que utiliza comunicación serial.

Industrial Shields. Industrial Shields. es una empresa que ha tomado bastante importancia en el mercado de instrumentación de procesos europeo, con su innovación y búsqueda de equipos PLC más flexibles y a un mejor costo.

Nace en 2012 bajo las ideas de Albert Prieto, desde entonces se han dedicado al diseño, fabricación y comercialización de PLCs basados en la filosofía Open Source Hardware, la mayoría de sus equipos se basan en placas de Arduino. *“Por lo tanto Industrial Shields es la marca que identifica el hardware libre que a la vez sigue parámetros de diseño, uso y seguridad industriales, uniendo lo mejor de dos mundos que hasta ahora avanzaban por separado”*

(Industrial Shields, Quienes somos, 2012, <https://www.industrialshields.com/es/about-us/>).

Kunbus. Es una empresa de fabricación de software y automatización cuyo propósito principal es el desarrollo de equipos para la industria basados en la filosofía Open Source, nació en Alemania en el año 2008, desde entonces se han especializado en comunicación industrial a través de buses de campo, Ethernet industrial y últimamente IoT (Internet de las cosas). Desarrollan soluciones inteligentes, confiables y económicas para redes de dispositivos de automatización, casi todos sus dispositivos funcionan basados en placas Raspberry PI.

Open Source (Código abierto). Según el portal Opensource.org, (2016), código abierto es un modelo de desarrollo de software basado en la colaboración abierta. Se enfoca más en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre. Para muchos el término «libre» hace referencia al hecho de adquirir un software de manera gratuita, pero más que eso, la libertad se refiere al poder modificar la fuente del programa sin restricciones de licencia, ya que muchas empresas de software encierran su código, ocultándolo, y restringiendo los derechos a sí misma.

Software con propietario (Software y dispositivo cerrado). Según el portal definicionabc.com, Este es el tipo de software más conocido a nivel mundial, puesto que el usuario tiene posibilidades limitadas de usar, modificar o redistribuir, y por lo general son productos que tienen costos para adquirir sus licencias.

No se ha llegado a un consenso sobre el término correcto para definirlo, sin embargo, en este proyecto hablaremos de él como software o sistema cerrado, puesto que su arquitectura no permite que un usuario pueda acceder a su código, y para acceder a él, se debe pagar por una licencia y solo puede usarlo en un contexto restringido, por lo que no puede ser modificado ni perfeccionado en su funcionamiento, como tampoco ser distribuido a otros destinatarios

Microframework Flask. Se trae a colación el concepto desarrollado para un artículo previamente escrito por el mismo autor del proyecto (Diseño y aplicación de un sistema bajo arquitectura cliente servidor para la activación de riego automatizado en Arduino a través de Twitter, página 5, Autor, 2018):

Microframework es el término que ha sido usado para referirse a Frameworks minimalistas de aplicaciones web, en contraste con Frameworks Full-Stack. Un microframework carece de la mayoría de funcionalidades que es común esperar en un Full-Stack, (como lo es Django) tales como manejo de cuentas, autenticación, autorizaciones, roles, permisos, abstracción de Bases de datos a través de un mapeo de objetos relacionales, motores de plantillas web, entre otras.

Las ventajas de tener un microframework se hacen mayores cuando la tarea de programar se vuelve simple, al ser escalable, Flask también permite añadir módulos, servicios o librerías, puesto que por defecto no está atado a ningún motor de plantillas o bases de datos, inclusive deja el manejo de excepciones a gusto del programador.

Pirámide de la automatización. Es un esquema que representa las capas básicas que relacionan todos los agentes que actúan en un proceso de automatización industrial, puede relacionarse con el modelo de la capa OSI, ya que en la base se encuentran los dispositivos físicos, y a medida que se escala en la pirámide, se encuentran los drivers, las redes, las interfaces y las herramientas que permiten la visualización, monitoreo, control y gestión de los datos e información. La pirámide de automatización planteada por la empresa mexicana de automatización SSIGSA DISEÑO, (2016) mostrada en *Figura 1, Gráfico de la pirámide de automatización básica de 4 niveles* da una visión más gráfica.

Figura 1
Gráfico de la pirámide de automatización básica de 4 niveles



Fuente: <http://ssigsa.com/images/piramide.png> (Consulta 20 de noviembre 2018)
Tomado de: SUMINISTROS, SERVICIOS E INGENIERÍA INTEGRAL DEL GOLFO S.A. DE C.V.

Según El Ingeniero de Sistemas, Cesar Augusto, en la blockchain social Steemit, (2018) En la parte baja de la pirámide se puede apreciar a Campo, y está bien representado como la base de la pirámide, porque es aquí donde reposa la mayor responsabilidad hablando en términos de confiabilidad y seguridad de datos.

Físicamente se refiere a los instrumentos de medición y acción del proceso automatizado. Aquí podemos mencionar los medidores de flujo, presión, densidad y temperatura, arrancadores suaves de bombas, actuadores de válvulas, solenoides, motores, robots, entre otros.

Posteriormente se aprecia a Control, donde se aprecia a mas maquinas PLC, que según el Ingeniero Cesar en Steemit: “Estos dispositivos reciben los datos de los instrumentos del área de campo. Y mediante algoritmos programados previamente,

procesan estos datos para luego dar una respuesta de manera automática a los dispositivos de salida o acción también en campo. Simultáneamente la información de estos datos recibidos, procesados y hasta las acciones tomadas, es de alguna manera enviada a consolas operaciones o computadoras ubicadas en el área siguiente que refiere a la supervisión”.

Seguido la tercera capa de abajo hacia arriba es Supervisión, se refiere a equipos donde se puede observar, supervisar y controlar las variables operacionales, tomando con punto de partida los datos de campo que fueron recibidos por los PIC's. A estos equipos se les dice Interfaz Hombre-Máquina (HMI) están ubicados por lo general más cerca del proceso automatizado en campo. En esta área se puede tener a los Sistemas de Supervisión, Control y Adquisición de Datos (SCADA - Supervisory Control And Data Acquisition) y los Sistemas de Control Distribuidos (DCS - Distributed Control System) estos últimos, normalmente, ubicados a una distancia mayor, dentro de lo que se conoce con el nombre de “Sala de Control” para procesos delicados de la industria.

Por último, en la capa superior se tiene a Gestión, por lo general son aplicaciones nativas al SCADA que almacenan los datos para luego ser procesados, almacenados y analizados de manera que se puedan tomar decisiones con respecto a ellos. Se considera más como una etapa administrativa, aunque por su impacto al momento de diagnóstico de fallas, estadísticas de tiempo de uso y vida útil de los equipos, además de la optimización del proceso es parte de un sistema de automatización.

Diseño metodológico

Estrategia metodológica

Es imprescindible saber cómo dar y recibir órdenes del PLC basado en arduino, y también conocer sobre los protocolos que se utilizan actualmente como interfaces de comunicación compatibles con el dispositivo cerrado AB, a fin de permitir que las entradas de este. que es el dispositivo que se encuentra instalado en las fábricas, sean las salidas del PLC arduino, y a su vez las salidas de AB. sean entradas para el PLC basado en placas de software libre.

Con la información recolectada sobre el funcionamiento de los dos dispositivos, y las tecnologías actuales en cuanto a los protocolos compatibles de interconexión, se sientan las bases para una 2a etapa, la cual encierra los conocimientos adquiridos y plantea un sistema en el cual los datos que son recolectados en campo, por los periféricos conectados al PLC AB. pueden ser procesados, guardados, organizados y mostrados a través de Arduino, con una interfaz desarrollada en software libre.

Cronograma de tiempos y tareas

El cronograma de tiempos y tareas que se tomó como referencia es el que se puede apreciar a continuación en *Figura 2, Cronograma presentado en la ficha técnica de radicación del proyecto.*

Figura 2

Cronograma presentado en la ficha técnica de radicación del proyecto

Etapas	Actividades	Semanas	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
		Tareas	MES 1			MES 2			MES 3			MES 4								
Etapa 1	Investigar documentación relacionada a la creación de una API	Principios de creación de api	X																	
		Diferencias entre API y SOA		X																
		Recursos para creación de API			X															
	Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen bradley y un PLC basado en arduino	Buscar documentacion relacionada al PLC AB				X														
		Recopilar y profundizar sobre PLC AB					X													
		Buscar documentacion relacionada al PLC Arduino						X												
		Recopilar y profundizar sobre PLC Arduino							X											
	Analizar cómo cada dispositivo PLC ejerce control sobre el sistema	Evaluar el sistema de recepcion de datos de PLC AB								X										
		Evaluar el sistema de entrega de datos de PLC AB									X									
		Evaluar el sistema de recepcion de datos de PLC Arduino										X								
		Evaluar el sistema de entrega de datos de PLC Arduino											X							
	Etapa 2	Diseñar una interfaz que permita administrar PLC Allen bradley desde un PLC basado en arduino	Levantamiento de requerimientos												X					
Elaboracion de casos de uso																X				
Desarrollo de especificación																	X			
Desarrollo de casos de prueba																		X		
Comunicación y divulgación de los resultados obtenidos de la investigación.		Sustentacion de resultados																	X	
		Divulgacion																		X

Fuente: https://docs.google.com/document/d/1KKM5pucbBJXj7taJgZmU_F6DU2B4ZB0qybuH8hhINgI/edit
(Tomado de: Ficha Api, Autor, Pág. 9)

Presupuesto de costos del proyecto

El presupuesto inicial del proyecto abarcaba los siguientes ítems:

- PLC M-duino, un millón cuatrocientos mil pesos (\$1'400.000).
- PLC Allen Bradley Micrologix 1100, un millón quinientos mil pesos (\$1'500.000).
- Arduino IDE, descarga gratuita en línea para todas las plataformas de sistemas operativos.
- Rslogix By Rockwell, versión gratuita.

Lo cual suma alrededor de 3 millones de pesos (3'000.000), esto sin tener en cuenta la mano de obra ya que al ser trabajo investigativo propio, es un monto totalmente asumido.

Debido al limitado presupuesto con el que se contaba inicialmente para el inicio y desarrollo del proyecto, y la inesperada ruptura de relaciones comerciales con un posible cliente que contaba con el dispositivo Micrologix 1100 de Allen Bradley, las alternativas con las que se pudo dar solución y un claro acercamiento a las herramientas reales de trabajo fueron:

- PLC M-duino Reemplazado por una placa Arduino Mega 2560 de cincuenta mil pesos (\$50.000).
- PLC Allen Bradley Simulado por una versión de prueba gratuita de 9 días del software RS Emulator de la suite de Rockwell Automation.

Cabe resaltar un hecho importante que fue descubierto posteriormente al planteamiento, por el cual los requerimientos de comunicación específicos para este proyecto del dispositivo AB. estipulaban software RsLinx y RsLogix que contarán con el aval de un licenciamiento pago, este inconveniente combinado con la pérdida de acceso al equipo físico, propició a obtener en línea una versión de prueba de nueve (9) días de la suite de Rockwell Automation, la cual incluye Factory Talk View, RSLogix 500 Pro, RSLinx Classic Gateway y RSLogix Emulate 500.

Participantes del proceso

Los participantes principales de este proyecto son el Docente PhD. Guillermo Solarte, en acompañamiento como asesor principal del proyecto a lo largo del desarrollo, compartiendo conocimiento sobre ingeniería de software desde la creación de la ficha de radicación, hasta la culminación del presente documento; Magister Julio César Hernández, quien brindó dirección y consejo a través de la materia Proyecto de Grado II; Docente Carlos Alberto Ocampo, cuyo conocimiento impartido en las materia proyecto de grado I e Ingeniería de Software III, fue de suma importancia para el análisis y posterior elaboración del diseño; y el Ingeniero Electricista Mario Alberto Valencia, profesor e investigador del Sena seccional Dosquebradas, experto con más de treinta (30) años de experiencia en automatización e instrumentación industrial en la región, Docente del curso de APLICACIÓN DE LOS PLC EN LA AUTOMATIZACIÓN DE PROCESOS INDUSTRIALES, que fue necesario para entender el funcionamiento de los PLCs, y quien impulsó la idea principal que le dio vida a la investigación.

Procedimiento

En aras de diseñar un sistema que permita la comunicación entre estos dispositivos que tienen 2 tecnologías completamente diferentes, lo más importante es dividir el proyecto en 2 etapas, una primera etapa que permita conocer cómo dar órdenes de entrada y salida al dispositivo que es conocido, PLC AB (Allen Bradley), estudiar su funcionamiento como maestro del sistema, y una segunda que permita analizar y diseñar posibles formas de transmitir datos entre el y uno equipado con un sistema abierto que permite múltiples formas de conexión.

Etapas 1: Investigación y captación de información.

En esta etapa se realiza la disposición y recolección de la información necesaria para el planteamiento de este proyecto, que abarca desde la conceptualización de herramientas y arquitecturas de actualidad para la creación de interfaces multiplataforma, hasta el reconocimiento de los ítems necesarios para el entendimiento tanto del PLC Allen Bradley como del M-Duino. Se divide en 3 actividades principales, la cuales se describen detalladamente a continuación:

Actividad 1, Investigar documentación relacionada a la creación de una API, SOA e interfaz web:

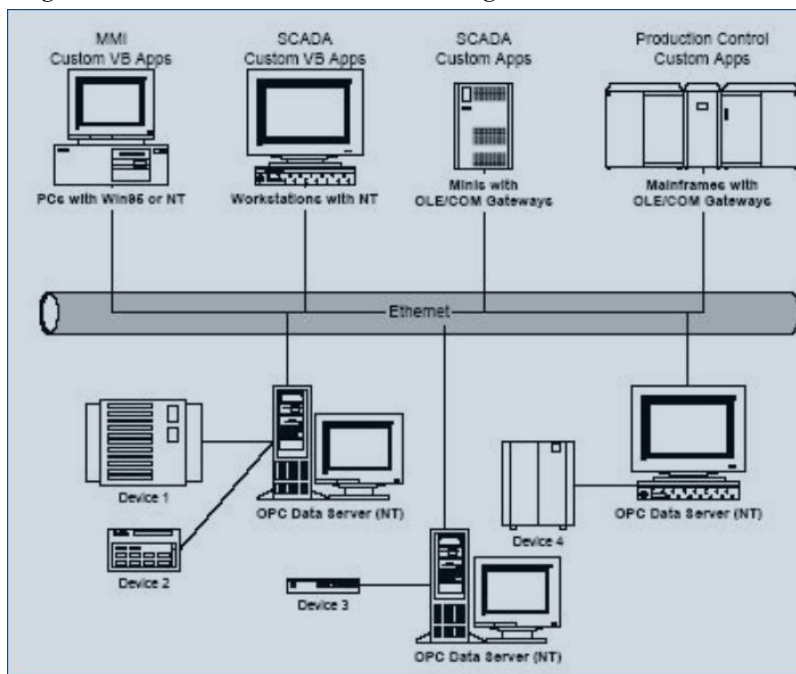
Al comenzar el proyecto no había mucha claridad a la hora de la etiqueta que se le daría al tipo de arquitectura del software que se diseñaba para encargarse de permitir una conexión multiplataforma, en este caso entre RSLogix 500 y Arduino IDE, sin embargo con el paso del tiempo, las horas de estudio y recolección de datos en el curso, fue posible descubrir otro estándar de comunicación que ha sido desarrollado en el campo del control y supervisión de procesos industriales, llamado OPC (OLE for Process Control) Según la fundación OPC (2018), OPC es un protocolo basado en una tecnología Microsoft, que ofrece una interfaz común para comunicación la cual a su vez permite que componentes de software individuales

interactúen y compartan datos, permitiendo que un dispositivo de hardware del nivel de planta de producción actúe como servidor de cualquier aplicación que se programe para leer/escribir cualquier variable que este ofrezca.

OPC está basado en OLE, software de tipo cliente pesado de Microsoft que permite crear un archivo principal, al cual se puede hacer un referencia de datos donde todo cambio posterior en el archivo principal se refleja en el documento donde se hizo referencia. A pesar de que los grandes fabricantes como Allen Bradley han incluido OPC en sus productos, y que hace posible la extracción de datos desde el dispositivo Micrologix 1100 a una hoja en excel como lo veremos más adelante, se hace necesaria la creación de un servidor programado como OPC que sea externo a la Suite de Rockwell Automation, como lo veremos a continuación en *Figura 3, Integración de sistemas en un entorno heterogéneo se torna sencillo a través de OPC:*

Figura 3

Integración de sistemas en un entorno heterogéneo se torna sencillo a través de OPC



Fuente:

https://www.google.com/search?biw=1366&bih=638&tbm=isch&sa=1&ei=wcz6W-m0F9L7zgLEgK_wDw&q=OPC+automatizaci%C3%B3n+wiki&oq=OPC+automatizaci%C3%B3n+wiki&gs_l=img.3...15567.16729..16885...0.0..0.149.714.0j5.....1....1..gws-wiz-img.Xi6I9QqCZDI#imgcr=s0OtDMwX1_sEoM

Tomado de: Google imagenes, Anónimo (Consulta 18 de Octubre 2018)

OPC plantea la interconexión de todos los dispositivos a través del protocolo ethernet, por lo que se plantea la creación de un servidor local en el micro framework de programación para python flask, el cual ha tomado gran fuerza en los últimos tiempos para la creación de API en python, ya ha sido trabajado y probado en conjunto con Arduino en un proyecto para la materia cliente servidor y es un excelente recurso a la hora de proyectos que tienen propuesto un servidor de características ágiles y livianas.

Actividad 2, Investigar y conocer el funcionamiento interno de un dispositivo PLC Allen Bradley y de un PLC basado en arduino.

PLC Allen Bradley.

A continuación se puede apreciar luego de una vista física en Figura 4, Imagen de un PLC Allen Bradley Micrologix 1100 Serie B, una descripción detallada de las características técnicas del PLC Micrologix 1100 y su controlador, tomado de la documentación oficial de rockwell Automation (2005):

*Figura 4,
Imagen de un PLC Allen Bradley Micrologix 1100 Serie B*



Fuente:

https://literature.rockwellautomation.com/idc/groups/literature/documents/pp/1763-pp001_-en-p.pdf

Tomado de: Manual oficial de ML1100 Serie B

El micrologix 1100 ha sido diseñado para monitorización remota y para aplicaciones que requieran bastante disponibilidad de memoria, pero requiere pocos puertos de entradas y salidas. Posee un socket Ethernet/IP de 10/100 Mbps para mensajería P2P (Peer to peer) y cobertura de conectividad para toda la familia de los productos PLC de la familia Allen Bradley bajo las limitantes del software producido por Rockwell Automation.

Contiene también un combo de puerto serial aislado RS-232 y RS-485, dos estándares de comunicación del protocolo serial, tecnología de baja velocidad de transmisión a través de un BUS de datos, con una tasa muy baja de pérdida de paquetes, características muy importantes en los procesos de recolección de datos físicos de los sensores y los actuadores en campo, esta ambigüedad de puertos en el dispositivo permite hosting de diferentes protocolos de red y de mensajería P2P y puede ser igualmente utilizada para la comunicación con otros controladores o PLCs.

Cuenta también con un contador de 40kHz rápido, para aplicaciones de robótica, una entrada rápida implica cambios de gran velocidad que no cualquier sensor detecta, en caso de que un PLC no tenga esa entrada, las señales de los sensores entran en rebote (Hay confusión de información).

Cuenta con 2 salidas de alta velocidad 40kHz PTO/PWM La cuales una tecnología modulación por ancho de pulsos en un ancho de banda, la cual sirve para trabajar señales analógicas, y resulta una forma eficiente de comunicar señales analógicas con bajos recursos.

Tiene instalada una memoria RAM de hasta 4 Kb para el Manejo de datos en tiempo de ejecución, lo que significa un máximo de 4 Kb como tamaño de datos de los datos análogos de todas las I/O, y también tiene una memoria ROM de 4 Kb que es un tamaño de memoria destinado a almacenar datos de usuario

La Cantidad de datos que puede ingresar simultáneamente es un máximo de 128 Kb y el máximo para recibir la información de los datos es de 64 Kb, lo cual representa bastante espacio teniendo en cuenta que tiene pocas entradas y salidas.

De las características que más destacan se encuentra el servidor Web integrado del equipo que proporciona una interfaz para conectarse en la nube fácilmente, también el soporte de emails para notificaciones en tiempo real y comunicación con la máquina remotamente .

Palabras más palabras menos, es un dispositivo muy completo, en términos de robustez y captación de información en campo, aunque tiene una limitada cantidad de entradas y salidas, 10 entradas de señales discretas, 2 de señales análogas y 6 salidas de señales discretas, es ideal para aplicaciones pequeñas, ya que también tiene una pantalla PLC que funciona como HMI al permitir la interacción Hombre-máquina para modificar los datos e interactuar con el programa de control con la capacidad de soportar un servidor web local.

Hasta ahora han sido solo halagos para este poderoso controlador, no merece menos, sin embargo, el punto de quiebre está en los costos de adquisición tanto del hardware, que oscila alrededor de US\$1,300, mil trescientos dólares, alrededor de cuatro millones de pesos (\$4'000.000), como del software que se requiere para el aprovechamiento de esta herramienta, es toda una suite de utilidades, de las cuales hablaremos en detalle a continuación.

RSLogix 500. Es el software destinado a la creación de los programas lógicos del autómatas en el lenguaje de contactos propio de estos dispositivos, que ha sido denominado Ladder, o lógica de escalera. Tiene entre otras funciones, el verificador de proyectos que guarda una lista de errores, el selector de modos que permite la edición offline de la lógica sin afectar el funcionamiento del sistema, entre otras funcionalidades. Es un producto que ha sido desarrollado para funcionar solo en sistemas operativos de Windows. El licenciamiento de este software ronda los tres millones quinientos mil pesos colombianos anuales (\$3'500.000).

RSLinx. Este software es un paquete de servidor de comunicaciones que permite la conexión de dispositivos controladores entre sí, y a su vez con la máquina que los controla. Puede soportar múltiples aplicaciones de software de dispositivos que trabajan simultáneamente en muchas redes diferentes. Según el portal Automation networks (Check out - RSLINX SOFTWARE), “RSLinx® clásico es el servidor de comunicación más ampliamente instalada en la automatización de hoy. RSLinx Classic Lite se puede utilizar ya sea como una aplicación o servicio. RSLinx Classic Lite, RSLinx Classic OEM o FactoryTalk Gateway como una aplicación proporcionan la capacidad de controladores, firmware flash y secuenciación por lotes del programa.”

Entre los firmware que proporciona este software, se encuentra DDE/OPC, el cual ha sido desarrollado para permitir la comunicación y transmisión de datos con otros software de Microsoft que han sido dotados con algoritmos y librerías OLE, como lo es Microsoft Excel, de esta funcionalidad hablaremos más adelante al detalle. puesto que es la herramienta de la cual se toma mano para el diseño de la interfaz. El licenciamiento de esta herramienta está incluido en el paquete RSLogix 500

RSEmulate: Este software es en esencia un simulador de hardware, en este caso del PLC micrologix 1100, el cual es utilizado para hacer prototipos y pruebas de arquitecturas diseñadas en el lenguaje ladder, y testear el diseño antes de realizar la descarga en el equipo, con algún error que pueda generar daños físicos en el PLC o en el sistema completo. Este es un software opcional, el cual puede ser o no adquirido para el correcto funcionamiento del equipo, su licenciamiento ronda los cinco millones de pesos (5'000.000) anuales.

RSView (SCADA): Es el software que está en la punta principal de la pirámide de la automatización, es el software que permite la exportación de la información a bases de datos o sistemas de gestión, así como el control y el monitoreo de la funcionalidad tanto del hardware de control, como de cada uno de los sensores y actuadores del sistema, permite la visualización de datos e informes programados, ya sean datos históricos o datos en tiempo real. El costo del licenciamiento de esta herramienta

Como ingeniero de sistemas, este es el tipo de software que se pretende diseñar y adecuar de una manera más libre y abierta, usando herramientas comunes como Excel y libres como python, que permitan la personalización de ambientes que funcionen en entornos de mayor libertad, que está también ligado al trabajo de adquisición del conocimiento para programar y desarrollar las tareas que implica la adecuación y mejora de una herramienta tan ampliamente utilizada por el sector eléctrico y electrónico, traída a la filosofía open source.

Debido a que el Micrologix 1100 es un equipo que está de fábrica optimizado para la comunicación a través de Ethernet pero con dispositivos de su misma marca, se busca aprovechar esta funcionalidad a través de el protocolo OPC que permite la interconexión con herramientas como Microsoft Excel, y de esta manera expandir las posibles funcionalidades de comunicación a las barreras que la imaginación y pericia del diseñador lo lleven.

El licenciamiento de este software depende de la versión que el proyecto requiera, para una sola estación de trabajo local sin posibilidad de comunicación a la

internet, está alrededor de los seis millones de pesos colombianos (6'000.000) anuales, y su costo incrementa exponencialmente al querer agregar estaciones de trabajo o conexión a otras redes, puede llegar a los treinta millones de pesos colombianos (\$30'000.000) anuales.

PLC Industrial Shields.

En las próximas páginas se encuentra un retrato de las características generales y detalladas de la familia de productos M-Duino, que han sido tomadas de la página oficial de Industrial Shields, (2018), teniendo en cuenta que todos los dispositivos de esta rama solo se diferencian por la cantidad y las características de las entradas y salidas de sus dispositivos, también se ha tomado como base un manual de usuario para todos los dispositivos M-Duino, de la misma empresa, Industrial Shields (págs. 10 - 16), que puede ser encontrado en las citas como “Manual de Familia M-Duino 23 de marzo”. Como se ha mencionado antes, esta familia ha sido escogida por su conectividad con redes Ethernet.

En este caso se toma la referencia del M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS, el cual se puede apreciar seguido a este párrafo en *Figura 5 Imagen de M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS*.

Está especialmente diseñado para su uso en un entorno profesional y cuenta con 13 entradas y 8 salidas (dispositivo de similares características al AB. que cuenta con 10 entradas y 6 salidas), de las cuales 7 entradas y 3 salidas son digitales configurables por software, cuenta con puertos de conexión para comunicación tipo Ethernet, USB tipo B, puerto I2C que ofrece la posibilidad de expandirse con 127 módulos y gobernar hasta 6604 E/S en modo maestro esclavo, además de módulos adicionales de sensores, o de expansión de la misma marca. También cuenta con dos puertos seriales, un RS-232 y un puerto HALF/FULL Duplex RS-485. Soporta protocolos TCP/IP, Modbus TCP y Modbus RTU y tiene la capacidad de comportarse como un Web server, lo que le ofrece una gran flexibilidad y control.

Admite 3 puertos de conexión TTL que permiten tarjetas propiamente electrónicas que funcionan a bajo voltaje y también un (1) Puerto SPI externo que usa MOSI,MISO, SS pines,tecnologías propias de de Arduino que brindan la capacidad de crear redes maestro-esclavo entre sus dispositivos.

Tiene la habilidad de funcionar en RTC (Real Time Clock) para que la comunicación con los actuadores sea instantánea en cuestión de milisegundos, tanto para generar los cambios en el sistema,como para ver las variaciones en el, tiene un alto nivel de amperaje con un consumo máximo de 1.2A,lo que le da la posibilidad de administrar sensores de proximidad los cuales consumen bastante energía. Soporta archivos de Arduino IDE de hasta 256 kB en su memoria flash, cuenta también con una memoria EEPROM, la cual funciona como ROM de 4 kB y una SRAM de 8 kB que funciona como una memoria RAM estática.

Figura 5

Imagen de M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS



Fuente:

https://www.industrialshields.com/es_ES/shop/product/is-mduino-21-m-duino-plc-arduino-ethernet-21-i-os-analog-digital-plus-3?category=1

Tomado de: Catálogo virtual de la empresa Industrial Shields

Su memoria Flash de datos de 256 Kb, duplica a la del PLC Allen Bradley, y puede ser programado a través del puerto USB, o inclusive remotamente a través del puerto Ethernet, esto ofrece acceso inmediato a la programación, mantenimiento y control desde cualquier lugar que cuente con una conexión a internet, tiene un costo de 196,00 €, alrededor de setecientos diez mil pesos colombianos (\$710.000).

Para su programación se emplea Arduino IDE, el cual es un entorno gratuito de programación que cuenta con gran soporte a nivel mundial, y para el cual Industrial Shield ha desarrollado las librerías necesarias para que sus equipos trabajen transparentemente, y solo es necesario seguir unos cuantos pasos de un pequeño tutorial que la empresa ofrece gratuitamente para la descarga y configuración correcta de todos los dispositivos.

Actividad 3, Analizar cómo cada dispositivo PLC ejerce control sobre el sistema.

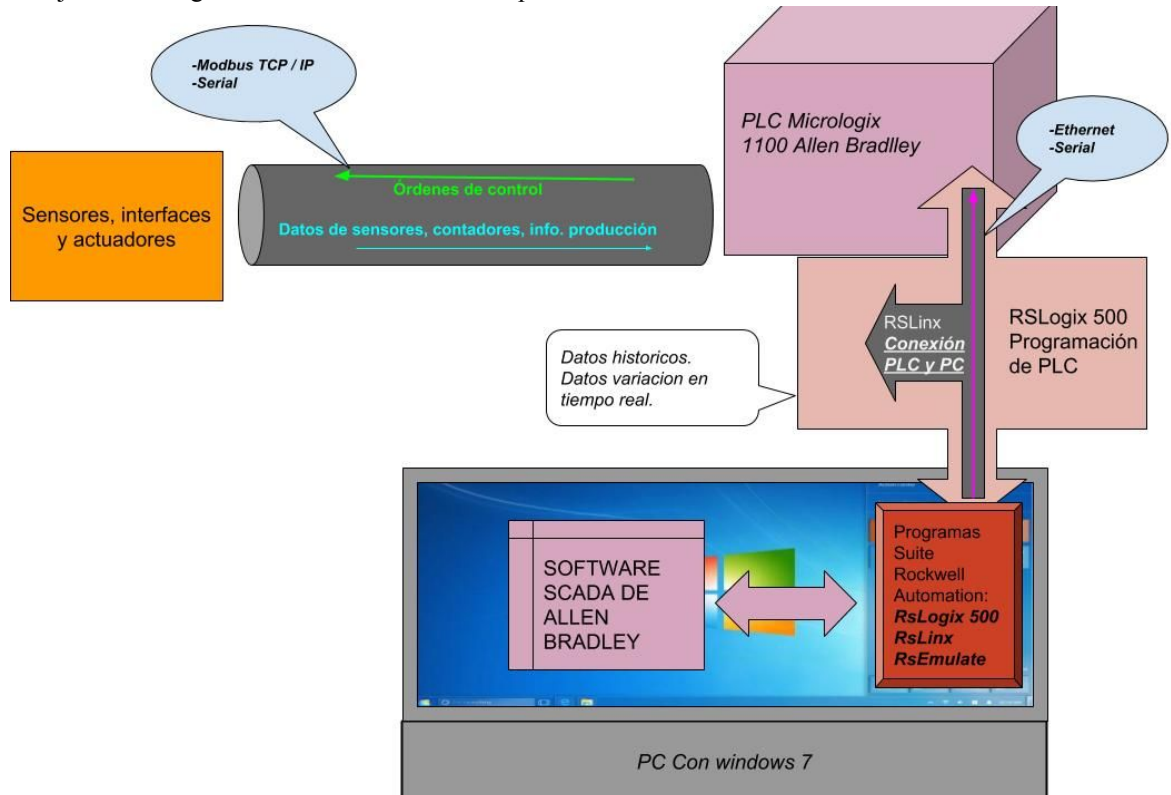
PLC Allen Bradley. Para tomar control del sistema, después de que todas las conexiones físicas han sido instaladas, el PLC Allen Bradley debe ser configurado en la plataforma RSLinx, la cual permite la comunicación entre los dispositivos de control a través del protocolo Ethernet.

Luego de que se selecciona la comunicación entre la tarjeta del controlador del cual se van a extraer los datos el PLC y el computador o la pantalla táctil con la que se va a descargar el programa, se procede a descargar el programa lógico en lenguaje ladder que ha sido programado en la plataforma RSLogix 500. El PLC recibe las órdenes a ejecutar en este programa a través del cambio de modo de *Offline* a *Descarga*, y una vez se vuelva a cambiar el modo a *Online* de inmediato comienza a ejecutar los datos de las órdenes de control que se indique en el código de lógica de contactos Ladder, sobre los actuadores del sistema, y recoger los datos que le entregan los sensores para su posterior análisis en el SCADA. En este punto es opcional el uso del software RSEmulate, debido a que se puede o no simular el proceso con el PLC, ya depende de la experticia y la planeación de quien ejecuta el proyecto.

Después de que el sistema ya esta probado y funcional, el control pasa al SCADA que se haya elegido, en la suite de Rockwell se ofrece RSView, sin embargo en este punto cualquier SCADA que tenga compatibilidad con este equipo puede tomar el mando. El SCADA se encarga de las ejecuciones de órdenes bien sea para la activación de máquinas, para el funcionamiento de una electroválvula, etc. Esto depende la naturaleza del proyecto. Paralelo a ejecutar las órdenes, el dispositivo se encuentra recibiendo datos en tiempo real de los sensores, si el proyecto cuenta con ellos, (por ejemplo un flujómetro que se encarga de la medición de flujo de determinado líquido que pasa por una área en determinado tiempo) el SCADA también según su programación estará almacenando información en determinada Base de Datos, para luego proceder a analizar los datos y generar informes gerenciales, de producción, administrativos, etc. A continuación en *Figura 6, Dibujo de visión gráfica*

del sistema controlado por PLC AB. se puede apreciar por medio de un diagrama la manera en cómo interactúan los datos en el sistema

Figura 6
Dibujo de visión gráfica del sistema controlado por PLC AB.



Fuente: <https://drive.google.com/open?id=1C8D2xmv8W7uToPJVAiUPzaLsy2U0It4socvwEzRwnTw>
Creado por: Autor

PLC Industrial Shields.

La primer gran diferencia que nos topamos al trabajar con estos dispositivos, en relación a los tradicionales, es que su lenguaje de programación es Arduino IDE, cuya adquisición es gratuita y tiene una estructura bastante simple, que divide la ejecución en dos bloques principales, `setup()` y `loop()`, la primer función que se ejecuta en el programa y solo una vez, es `setup()` donde se definen las variables pinmode (qué pines digitales son de entrada o salida) y se inicia la comunicación

serial. Loop() contiene todo el código que será ejecutado en cascada, donde se configurará el comportamiento de las entradas y las salidas

A continuación la *Figura 7 Ejemplo de programación de dos salidas y su activación* nos ilustra un ejemplo de la programación básica de 2 salidas, podemos apreciar en el recuadro marcado en la parte superior izquierda el ejemplo de código que se ejecuta para definir dos señales de voltaje como variables enteras Q06 y Q05, que más adelante son configuradas como salida en la función **setup()**, el bloque del medio, y en la parte inferior los comandos que activan una salida con señal de voltaje por 1000 milisegundos y otra que permanece apagada. Es un ejemplo pequeño de lo sencillo que es configurar entradas y salidas.

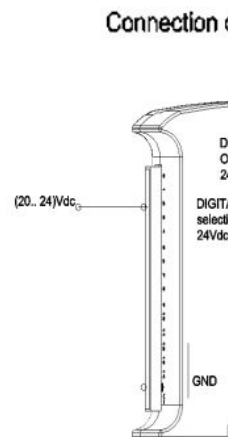
Figura 7
Ejemplo de programación de dos salidas y su activación

```
int Q06 = 3;
// select the Analog (0-10Vdc) / PWM (10 OR 24Vdc) /Digital
//(24Vdc)OUTPUT
int Q05 = 5;
// select the Analog (0-10Vdc) / PWM (10 OR 24Vdc) /Digital
//(24Vdc)OUTPUT

void setup() {
  pinMode(Q06, OUTPUT);
  pinMode(Q05, OUTPUT);
}

Void loop(){
  analogWrite(Q06, 254);
  digitalWrite(Q05, HIGH); // Relay ON
  // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second

  analogWrite(Q06, 0);
  digitalWrite(Q06, LOW); // Relay OFF
  // turn the LED off (LOW is the voltage level)
}
```



Fuente: (Consultado 1 Octubre 2018)

(<http://www.automaticaeinstrumentacion.com/es/downloads2/industrial-shields-albert-prieto.compressed.pdf>)

Tomado de: *Presentación modelo propio industria 4.0*, Albert prieto, 2016

Luego encontramos el bloque ***loop()*** donde el valor analógico 254 es escrito y asignado en Q05, y también es asignado el valor digital HIGH, por lo tanto Q05 es una salida abierta, y una vez que se ejecuta el código esta salida permanecerá activa por 1 segundo, usando la función *delay(1000)* cuyas unidades son milisegundos. Por otro lado, en Q06 es escrito por valor analógico el 0, y por valor digital LOW, por lo que al ejecutarse el código esta salida permanecerá inactiva y sin señal de corriente.

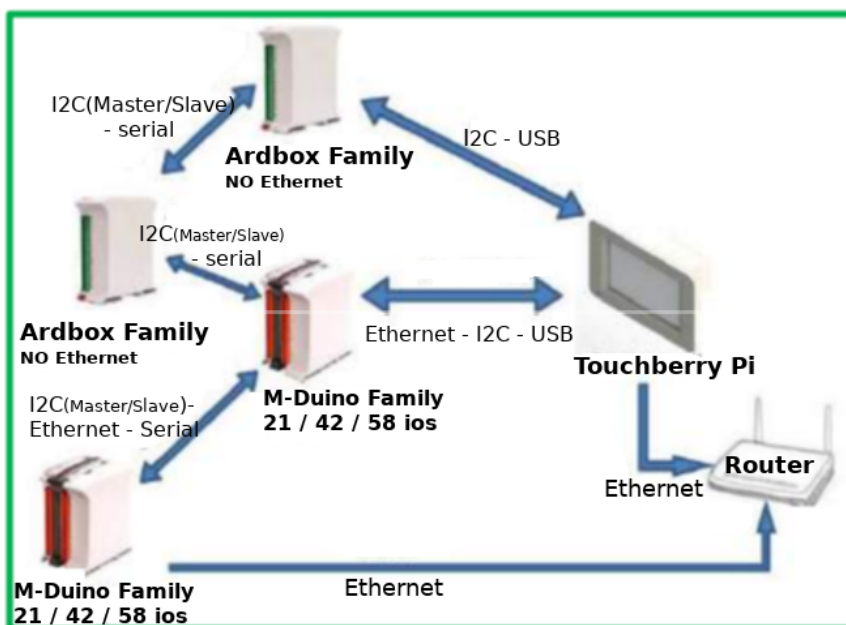
La segunda gran diferencia tiene que ver con el uso de los protocolos de comunicación de manera libre para el intercambio de datos físicos entre diferentes PLC, los cuales son: el protocolo I2C, protocolo propiedad de arduino el cual funciona sólo bajo sus dispositivos y la arquitectura maestro - esclavo, protocolo Serial, el cual es el más usado en la búsqueda de estabilidad, confiabilidad y seguridad en los datos sacrificando un poco la velocidad y rapidez entre ellos y, solo entre dispositivos M-Duinos, protocolo Ethernet.

La comunicación con los Panel Pc puede ser también a través de protocolo I2C, en este caso no es necesario que un dispositivo funcione como esclavo, protocolo USB, que ha tomado mucha fuerza en los últimos tiempos en la transmisión de datos de voz y video, y protocolo Ethernet solo con PLCs M-Duino, esto puede ser apreciado en la *Figura 8, Plano de conectividad de un sistema ejemplo con varios PLC, una pantalla y un router* se muestra cómo serían los protocolos de comunicación entre PLCs Arduino, I2C(maestro-esclavo) partiendo de los ArdBox, serial o ethernet entre M-Duinos. Entre los PLC y la pantalla podrían ser Ethernet, I2C o USB según el PLC. las conexiones de pantalla y PLC al router se harían a través de ethernet

Por último la comunicación con el router se limita solo a pantallas táctiles y a dispositivos M-Duino, debido a que los Ardbox no cuentan con puerto Ethernet. En caso de querer transportar datos a través de Ethernet usando un router, en una red de PLCs Ardbox debe existir un dispositivo M-Duino que actúe como maestro, y a su vez se comunique con un router a través de Ethernet, todas estas conexiones se pueden apreciar gráficamente a continuación.

Figura 8

Plano de conectividad de un sistema ejemplo con varios Plc, una pantalla y un router



Fuente: (Consultado 1 Octubre 2018)

(<http://www.automaticaeinstrumentacion.com/es/downloads2/industrial-shields-albert-prieto.compressed.pdf>)

Tomado de: *Presentación modelo propio industria 4.0*, Albert prieto, 2016

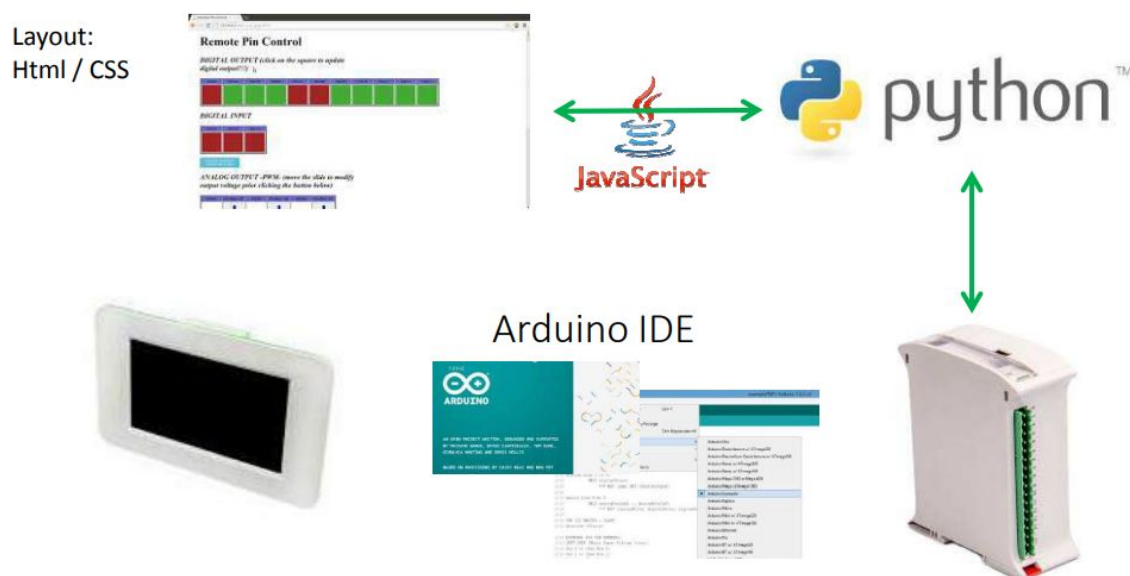
En cuanto a la programación lógica y la estructuración de los datos para que estos viajen transparentemente y sean visibles para el cliente final con características de usabilidad y facilidad de uso, se plantea que la comunicación entre el PLC y el panel táctil, que son los dispositivos que deben estar en la zona de producción o cerca de donde se encuentra instalado el automatismo, se comunican a través la librería “Industrial Shields” para Arduino que como fue mencionado antes, es totalmente gratuito y cuenta con soporte a nivel mundial en diferentes idiomas, oficial de Arduino y de comunidades que se han dedicado a investigar y aplicar en muchas áreas de la tecnología sistemas con estas placas.

Para la comunicación entre el PLC y un software de captación de datos, que puede albergarse en un servidor local o en la nube, la empresa proporciona un módulo de Python asociado que requiere sólo unas pocas líneas de código para usar las características del hardware del Ardbox o del M-duino. Cuando se importa un programa de Python, el módulo de Industrial Shields crea objetos de software para entradas analogica, digital, salida

analógica, digital, de relé y control de luz LED, cada uno de los cuales incluye las funciones de lectura o escritura de nivel inferior adecuadas mediante el manejo de transacciones detalladas.

Luego, de esto, la empresa recomienda el uso de Javascript para el tratamiento y procesamiento de datos para la nube, ya que posibilita que la página web sea lo menos estática posible, y viene de la mano con otros lenguajes de obligado aprendizaje, como HTML y CSS para completar “la ordenación y colocación de todos los elementos que componen una página web”(Nelly Natali,10 marzo 2009,<http://blog.nnatali.com/2009/03/10/css-que-es-un-layout/>). Es posible visualizar en el siguiente gráfico que permite ver cómo pueden interactuar los dispositivos con los lenguajes de programación. *Figura 9 Ilustración de los lenguajes de programación usados en una arquitectura con dispositivos de industrial shields.*

Figura 9
Ilustración de los lenguajes de programación usados en una arquitectura con dispositivos de industrial shields



Fuente: (Consultado 1 Octubre 2018)

(<http://www.automaticaeinstrumentacion.com/es/downloads2/industrial-shields-albert-prieto.compressed.pdf>)

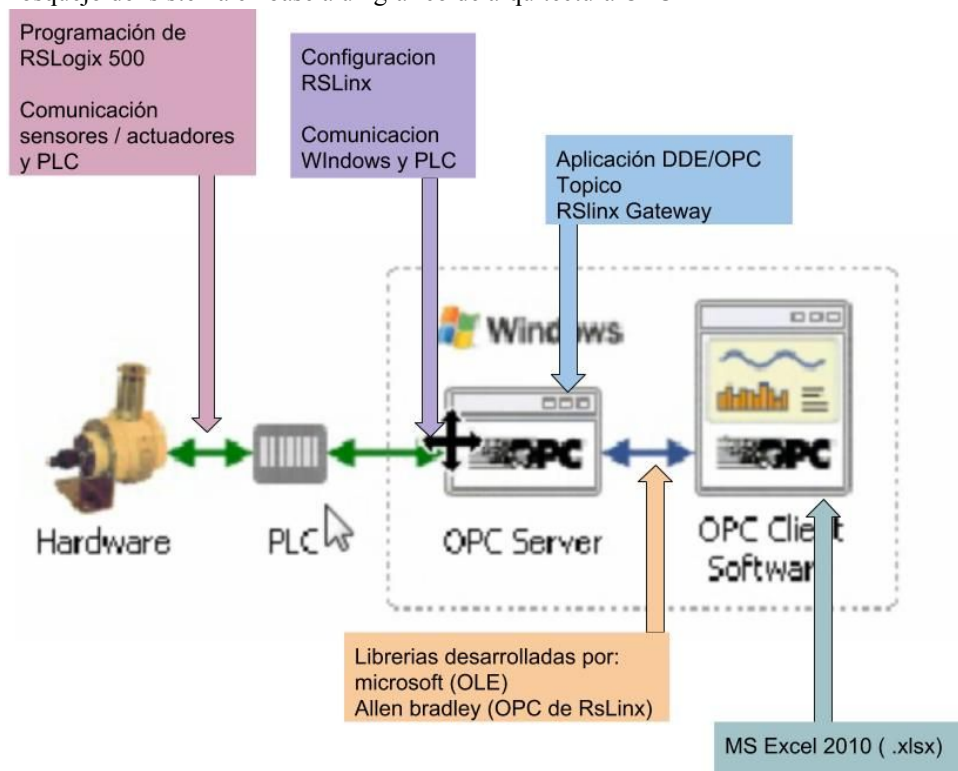
Tomado de: Presentación modelo propio industria 4.0, Albert prieto, 2016

Etapas 2: Diseño y divulgación.

Actividad 1, Diseñar una interfaz que permita administrar PLC Allen Bradley desde un PLC basado en arduino

Esta actividad fue planeada teniendo en cuenta los conceptos básicos de análisis y desarrollo de una interfaz, sin embargo con el desarrollo de la investigación, se descubrió el estándar OPC el cual surgió en 1995 con la unión de los grandes fabricantes de hardware y software industriales, que junto con Microsoft desarrollaron la primera versión de lo que sería el estándar mundial de protocolos de comunicación e interoperabilidad entre dispositivos de automatismos industriales. Es por esta razón que se decide planear la interfaz en base a esta arquitectura, la cual tiene su primer bosquejo graficado seguido en *Figura 10 Bosquejo del sistema en base a un gráfico de arquitectura OPC*.

Figura 10
Bosquejo del sistema en base a un gráfico de arquitectura OPC



Fuente:

https://drive.google.com/open?id=1mr3pUbPC7998LHjRuIBAXPeIAP69Hmny_XvYYuwPqv8

Tarea 1, Levantamiento de requerimientos:

La interfaz a desarrollar debe tener un módulo *Acceso_Datos_Sincronos* que permita leer, escribir y monitorizar variables que contienen datos de proceso actuales en el PLC Allen Bradley Micrologix 1100. Su principal uso será el transmitir datos de tiempo-real entre PLC AB. Y PLC M-Duino 21 I/Os plus, y otro dispositivos de control en un PC pantalla cliente. Se debe poder seleccionar explícitamente las variables o tópicos (elementos OPC) que quiera leer, escribir o monitorizar en el servidor desde el PLC AB, bien sea a través de la interfaz o usando el *Anexo D, Configurar un tópico OPC/DDE en RsLinx para obtener datos de un controlador Logix 500 y agregarlo a una hoja de cálculo de excel*, al final de este documento. A través de la interfaz OPC se establece una conexión al servidor M-Duino creando un objeto OPCServer, donde los datos de salida del Allen Bradley son las entradas de la interfaz, y contiene los métodos *navegar_direcciones* y *buscar_dato OPC* y sus propiedades, como el tipo de datos, o los permisos de acceso, los cuales se guardaran en un diccionario de datos .

Se debe poder escoger entre 2 modos, el primero monitorizar los cambios de los valores por el M-Duino, a fin de que si este detecta un cambio que supere límites establecidos, reaccione con cierto tipo de alerta o mensaje y gestione este problema ofreciendo el instante de muestreo (timestamp) y la calidad a los datos enviados. La calidad especifica si el dato es correcto (bueno), no disponible (malo) o desconocido (dudoso).

El segundo modo, una visualización en pantalla cíclica de los datos en busca de cambios para la supervisión en tiempo real, con una demora de actualización que no debe ser mayor a 2 segundos. Este tiempo de actualización es utilizado en el servidor para comprobar los valores cíclicamente en busca de cambios. A cada ciclo, el servidor envía sólo los valores cambiados al usuario.

Otro módulo que se propone es Acceso_Datos_Historicos, que debe permitir el acceso a datos ya almacenados. Los archivos históricos se deben recuperan de manera uniforme, desde el sistema de registros simple del Micrologix, hacia la interfaz OPC, como se puede apreciar al final del documento en Anexo D.

Se plantea la creación de un Objeto OPCH_Server en el servidor Arduino. Este objeto debe tener todas las interfaces y métodos para leer y actualizar datos históricos. Un segundo objeto OPCH_Browser está definido para navegar el espacio de direcciones del servidor Arduino.

Se proponen tres funcionalidades para la lectura de datos históricos en función de la generación de informes:

La primera es leer datos ‘crudos’ del archivo, que no ha sido procesados, donde el cliente define las variables y la cantidad que quiere leer. El servidor devuelve todos los valores archivados en cantidad especificada hasta el número máximo de valores especificado por el cliente .

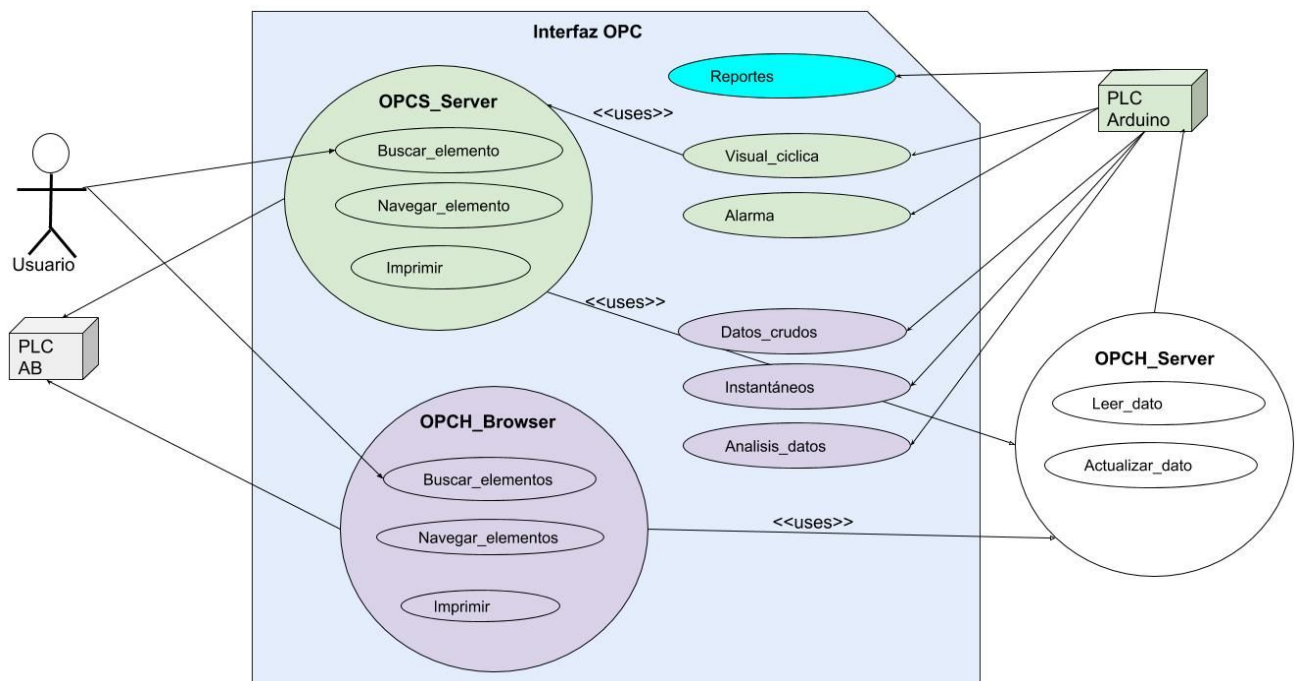
La segunda es de leer los valores de las variables especificadas para un periodo de tiempo determinado.

La tercer funcionalidad debe permitir que se hagan cálculos a los valores agregados de datos en la base de datos de históricos para los valores especificados en la ventana temporal, a razón de permitir análisis profundos a los datos con tratamiento matematico y estadistico.

Tarea 2, Elaboración de casos de uso

La elaboración de los casos de uso gráficamente apreciada a través de *Figura 11* Diagrama UML de los casos de uso, se hace de acuerdo al levantamiento de requerimientos hecho en la tarea anterior y se ve a continuación.

Figura 11
Diagrama UML de los casos de uso



Fuente: <https://drive.google.com/open?id=1Soi45kSRXP0P9KLJWjysPHs231Gzm85Zhv8zNmqTA9A>
Autor

Tarea 3, Desarrollo de especificación

Para la especificación del sistema se plantea *Tabla 8 Documentación de los actores que interactúan en los casos de uso*, donde se aprecian los actores que interactúan más adelante en *Tabla 9 Lista de drivers arquitectónicos*.

Tabla 8
Documentación de los actores que interactúan en los casos de uso

Actor	Caso de uso en los que participa	Tipo	Descripción
Usuario	2	Primario	Cliente que manipula el sistema
PLC Allen Bradley Micrologix 1100	2	Secundario	PLC que funciona como esclavo y se encarga de recoger los datos de campo
PLC M.Duino 21 I/O	7	Secundario	PLC que funciona como maestro y recoge los datos para administrarlos, analizarlos y mostrarlos

Tabla 9
Lista de drivers arquitectónicos

Tipo de driver	Id	Descripción del driver	Prioridad
Atributo de Calidad Disponibilidad	AC1	El sistema debe estar disponible un 98% del tiempo de uso	5
Caso de Uso	CU1	El sistema debe permitir buscar una variable con datos actuales almacenada en el PLC Micrologix	5
Caso de Uso	CU2	El sistema debe permitir la navegación hacia el elemento buscado en Micrologix a través de las herramientas OPC	5
Caso de Uso	CU3	El sistema debe permitir la impresión por pantalla de los datos seleccionados	5
Caso de Uso	CU4	El servidor PLC M-duino debe leer la información que se trae desde Micrologix	5

Caso de Uso	CU5	El sistema debe poder analizar datos históricos recibidos y entregar estadísticas	4
Caso de Uso	CU6	El sistema debe poder mostrar una búsqueda instantánea de datos históricos en determinado periodo de tiempo	3
Caso de Uso	CU7	El sistema debe permitir la visualización gráfica de datos cíclicos actualizados cada 2 segundos	3
Caso de Uso	CU8	El sistema debe permitir la configuración de alarmas bajo límites establecidos	3
Restricción	R1	El sistema no debe permitir la búsqueda de dos tipos de dato a la vez	4
Atributo de calidad Desempeño	AC2	El sistema no debe tardar más de 1 minuto en hacer una búsqueda de un conjunto de datos históricos	3
Atributo de calidad Desempeño	AC3	El sistema no debe tomar más de 2 segundos para actualizar los datos en tiempo real	3
Atributo de calidad Usabilidad	AC4	El sistema debe ser de facil manipulacion para el usuario final	5
Restricción	R2	La interfaz debe estar programada en python 3, usando la librería OPENPYXL para la comunicación con Excel, y SOCKET para la comunicación con Arduino	4

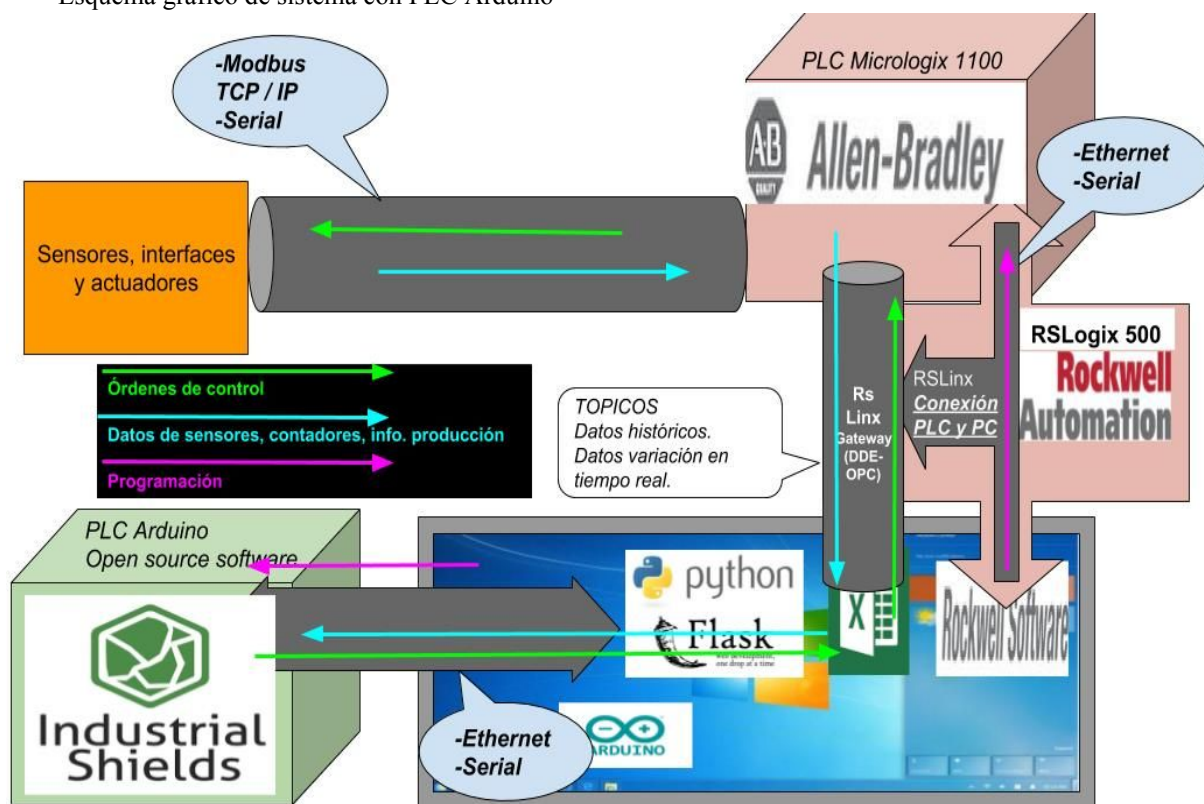
Resultados

Los resultados más notables de esta investigación pueden apreciarse en la etapa 2 del procedimiento, puesto que el objetivo principal de este proyecto fue el de analizar los elementos de un sistema de automatización industrial actual, manejado por el común Micrologix 1100, visto desde la perspectiva de un Ingeniero de Sistemas, para así de esta manera diseñar una interfaz de conexión entre el sistema con este robusto equipo, y un dispositivo que está basado en una filosofía de software libre.

A pesar que las tareas de la etapa 2 describen desde el análisis de los requerimientos hasta el diseño de la interfaz, lo hacen de una manera conceptual, visto desde la Ingeniería de Software. Es por esto que para tener una vista más global en función de la presentación del uso de todas las herramientas presentes, buscando la integración de conocimientos técnicos del área eléctrica de la instrumentación y automatización con área de sistemas y desarrollo, se

muestra a continuación la *Figura 11, Esquema gráfico de sistema con PLC Arduino*, el cual es un esbozo más completo de todo el ciclo de flujo de datos que se plantea, desde las conexiones físicas con los sensores y actuadores, como se pudo apreciar anteriormente en la Figura 6, pero en este caso agregando el túnel de conexión OPC que brinda RSLinx hacia una hoja de cálculo en excel.

Figura 11
Esquema gráfico de sistema con PLC Arduino

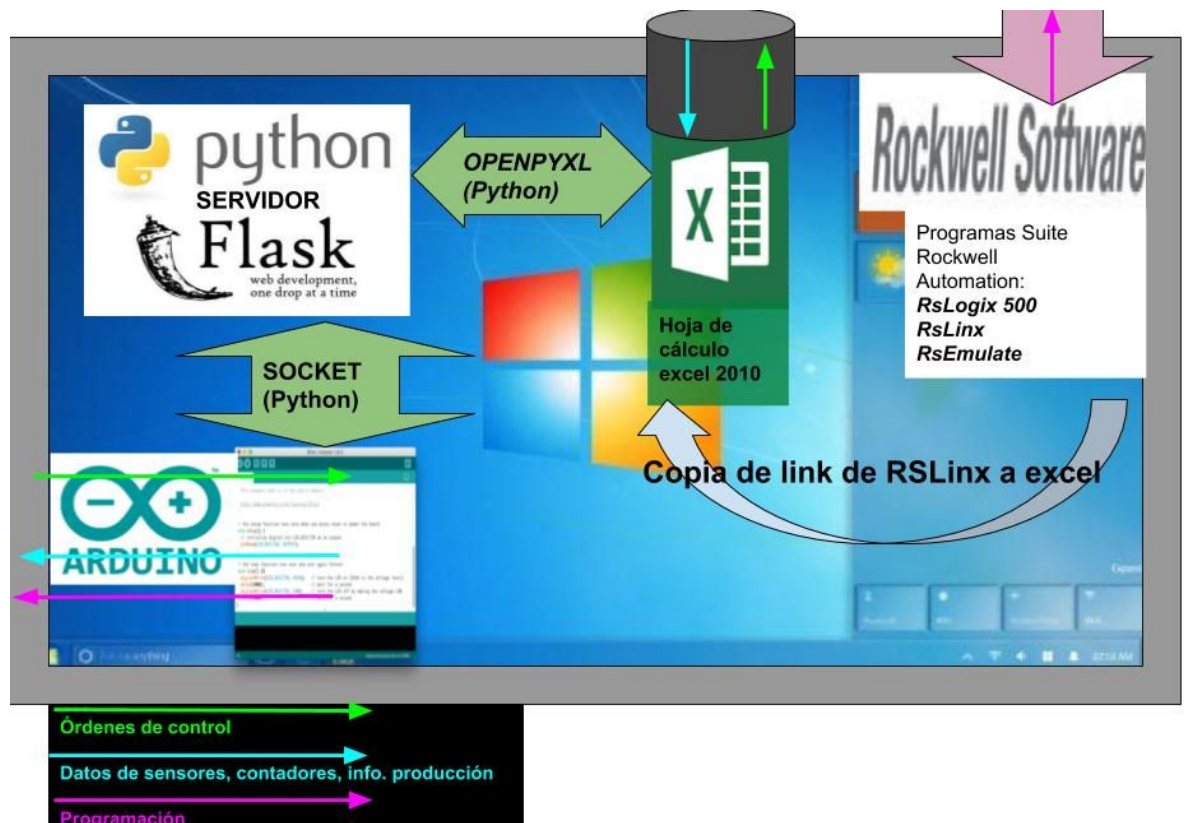


Fuente: <https://drive.google.com/open?id=1JpLO994YGAXp8--BG-rqkr5qOZLKT3ApebOtgSFRhH0>
Autor

Cabe resaltar que este puede no ser el esquema final de un producto terminado, esto responde a la corta experiencia, idoneidad y sabiduría de todos los elementos que componen un sistema industrial automatizado, pero es una clara muestra de que es posible utilizar herramientas comunes, como lo es Microsoft Excel, y junto con herramientas libres entrar en el campo del desarrollo de sistemas administrativos y de gestión de la información, cerca a lo que puede ser un SCADA, es posible apreciar este modelo más de cerca en *Figura 12*,

Esquema gráfico acercado de los elementos de desarrollo, en función de los elementos de desarrollo y programación necesarios que han sido descubiertos y tridosha colación en el desarrollo de esta investigación.

Figura 12
Esquema gráfico acercado de los elementos de desarrollo



Fuente: https://drive.google.com/open?id=1F1_qU8LE3dcFcDPpFcirMoABXcypsIRTHTHuoPKXZw0
Autor

Es muy bien recibido por el sector de automatización local este tipo de iniciativas y proyectos, pues es visible la falta de herramientas y elementos que permitan una mejora en la optimización de recursos y procesos de la mediana y gran industria de la región cafetera, que se ajusten a las necesidades y los bolsillos de los empresarios, y que a su vez fomenten el desarrollo de proyectos de innovación y emprendimiento regional.

Discusión

Finalizado el análisis y el diseño de la interfaz para la interconexión y transmisión de datos entre diferentes tecnologías de PLC, lo primero que se trae a colación es el planteamiento de la interfaz, puesto que al principio de este proyecto, se tenían ideas muy nubladas sobre las tecnologías que usa la industria de la automatización. Uno de los retos más interesantes fue la formulación del proyecto, debido a que se tenía desconocimiento sobre el tipo de interfaz que debía usarse, API, SOA, y otros modelos fueron ideados, y tratados de llevar a la conceptualización, sin embargo, fue solo hasta el desarrollo completo del proyecto que se pudo encontrar el estilo de arquitectura que se ha usado en esta misma industria hace unos años, OPC, que es ahora un estándar mundial para todos los grandes fabricantes de equipos de automatización y que tiene bases muy sólidas desde 1995, es lo que ha permitido la interoperabilidad entre equipos y software de diferentes fabricantes.

El segundo factor a discutir tiene que ver con los límites y el alcance que tuvo el proyecto en su formulación, si bien este proyecto nace de un problema real que se presentó en una empresa, y su componente principal PLC Micrologix 1100 fue el mismo que se pensó para este proyecto, se dificulta demasiado el trabajo en base a este equipo, pues su acceso es costoso, y para el tiempo que el proyecto estaba tomando forma, se habían roto las relaciones comerciales y ya no se posee acceso al equipo ni al software que se necesita para su programación y funcionamiento, aunque este inconveniente tuvo solución por medio de una versión demo por 7 Días, se pudo constatar lo cerrada que es la arquitectura de la empresa Rockwell Automation, y los altos costos que puede generar el acceso completo de las herramientas que permiten utilizar un recurso como este, así como la complicada labor de cotizar los costos. El distribuidor oficial, Melexa, ubicado en el sector industrial la popa, nos brindó la información de costes que se puede apreciar en el desarrollo de la metodología, sin embargo, fue tras una espera de alrededor mes y medio que fue posible tener una reunión con el encargado para el Eje cafetero, debido a la complejidad que exige nuestro proyecto en torno a la programación y el conocimiento del equipo. Esto también da una visión de la espera que puede generar plantear un proyecto con una tecnología como la de Allen Bradley.

Otro hecho que ha generado un contraste ha sido la elección del protocolo a utilizar para la comunicación de los equipos, si bien Ethernet ha sido un protocolo que ha tomado mucha fuerza en los últimos años para el desarrollo de redes industriales de velocidad mas rapida para la transmisión de datos, y esto ha tocado la puerta del mundo de la automatización, con nuevos equipos que incorporan puertos Ethernet, se desconocía totalmente el estado del arte de este sector de la industrial, y a través de esta investigación se ha podido constatar que a pesar del avance de nuevos protocolos, los protocolos Serial son aún hoy en día los mas usados para la transmisión estable de datos entre equipos o entre PLCs y sensores y actuadores. RS232 y RS485 son aun protocolos vigentes en redes de automatización industrial, pues garantizan la fiabilidad y estabilidad de los datos sin importar los ruidosos o toscos ambientes donde se encuentren instalados, y son los más usados cuando lo que se busca es poca caída de paquetes en la transmisión de datos críticos, como puede serlo cualquier proceso que involucre recursos de una empresa. Es por esto que se encontró mucha documentación sobre la comunicación entre Arduino y el software de Rockwell, sin embargo las librerías que se ofrecían solamente establecen comunicación a través de protocolos seriales, y este es el motivo por el cual tuvo que ser añadida una herramienta como Excel, ya que por la colaboración de Microsoft con la industria de la automatización, el protocolo OLE permite la interacción entre productos Microsoft y Allen Bradley. Se pudo constatar lo poderosa que es esta herramienta de ofimática, sin embargo hace que el planteamiento de este proyecto resulte en muchas rutas adicionales para los datos, lo que puede retrasar bastante el tiempo de respuesta y reacción del sistema, en comparación con una arquitectura con conexión por protocolos seriales.

Como se puede observar en los anexos, Industrial Shields es solo una mas de las empresas que ha incursionado en el mundo del desarrollo libre para elementos de automatización de procesos industriales, ya que es un nuevo mercado emergente en el que se han sumado varios fabricantes, como lo son Controllino, Kunbus, entre otras. La escogencia de esta empresa fue basada en la tecnología de las placas bases que usan, puesto que por conocimiento y experimentación propia para proyectos de otras materias y que fueron incluidos en esta investigación, se posee una experiencia previa y un conocimiento sobre el

funcionamiento y la programación de Arduino más allá de las placas basadas en Raspberrypi de la industria alemana. Esto no quiere decir que Industrial Shields sea la compañía que tenga que usarse al pensar en un proyecto como estos, porque como se pudo descubrir a medida que este proyecto avanzó, trabajar con una placa Raspberry puede tener más ventajas a la hora de buscar interoperabilidad entre los sistemas, al ser una placa que funciona bajo sistemas operativos Linux y que tiene más capacidad de procesamiento, y son muchas las opciones que ambas empresas ofrecen en cuanto a productos y servicios.

Lo que diferencia grandemente a estas dos empresas es el modelo de arquitectura para la construcción de los módulos PLC, Industrial Shields ofrece sus dispositivos compactos, listos para ser programados y conectados a una fuente de energía eléctrica, por otro lado Kunbus plantea soluciones con equipos modulares de 3 partes, Módulo Central, que controla las operaciones, un módulo de entradas y salidas, que es la expansión que permite conectar IOs, y modulo fuente de poder que entrega la energía necesaria para el funcionamiento del sistema de control. Estos 3 son los elementos de hardware que se necesitan para crear un sistema que se programe y conecte a los sensores. Las connotaciones que trae esta diferencia son dos, en relación a capacidad y precio. Por un lado, como es bien sabido por las personas que integran sistemas de control, los PLC cuya arquitectura es modular son equipos que por lo general superan por mucho en capacidad de procesamiento de procesos y entradas salidas, a dispositivos compactos. Esto resulta en la segunda connotación que tiene que ver con que un sistema con PLC modulares resulta más costoso que trabajar con Compactos, por lo que su escogencia debería depender del tamaño del proyecto que se quiere abordar.

Otro punto de discusión para este sistema es el hecho del planteamiento inicial utilizando otro dispositivo PLC Arduino, pues se pudo constatar que seria innecesario agregar otro PLC Arduino para dar órdenes a un sistema básico que solo cuente con 1 PLC Micrologix, ya que con el desarrollo de una interfaz que integre los elementos vistos anteriormente, como lo son la conexión OPC a un sistema en python a través de excel, es suficiente el poder de procesamiento que se necesita para generar informes, alarmas y órdenes. El agregamiento uso de un dispositivo arduino de control sería imperante en el momento que se desee adecuar un sistema que cuente con varios PLC conectados en red. En

este caso sí traería muchas ventajas la implementación de un equipo que se dedique solo a dar órdenes y administrar los datos recibidos, basado en filosofía de software libre. Este supuesto se hace después de constatar lo poderoso que es el equipo Micrologix 1100, en cuanto a capacidad de procesamiento.

Junto a la conclusión anterior se tiene la nueva concepción de que a la hora de proponer nuevos proyectos, se busque seleccionar solo equipos de programación libre, sin la necesidad de agregar dispositivos que se dediquen solo al control de las entradas y salidas y otro al control del sistema completo, pues los dispositivos industriales de arduino cumplen con toda la normativa requerida en términos de seguridad y rendimiento Europeo, y que una adecuación como la vista en todo el desarrollo del procedimiento sería óptima en el momento que solo se quiera adecuar un sistema que ya está funcional, explícitamente como el objetivo general de este proyecto lo indica.

Anexos

Historia de los PLCs

Nos remontamos a finales de los años 60's, momento en el que la industria automotriz a nivel mundial, se encontraba en gran auge comercial y tecnológico, esto causó necesidad a un mayor progreso y desarrollo por parte de los fabricantes de dispositivos de control, fue así como comenzó el desarrollo de los microcontroladores por todo el globo terráqueo.

Para llegar al dispositivo PLC que se conoce ahora, fabricantes como Bedford Associates Inc, (quien fue el primer constructor de lo que hoy se conoce como Controlador Lógico programable) Allen bradley, quien posee PLC™ como marca registrada propia y otros cuantos, comenzaron una carrera por desarrollar un nuevo dispositivo que permitiera ofrecer soluciones cada vez más innovadoras a los requerimientos de la industria automotriz en Estados Unidos, llenando la década de los 70's con avances importantes en el campo de la interacción hombre máquina, espacio en disco duro, lenguajes con más y potentes instrucciones, entre otras mejoras.

Estado del arte

Desde entonces han surgido más fabricantes como Siemens, Delta, Omron, ABB y muchos otros, quienes se han dado la tarea de crear dispositivos cada vez más robustos con funcionalidades mejoradas, sin embargo, debido a que cada fabricante puede tener procesos y objetivos diferentes, la unificación y estandarización de los lenguajes de comunicación en todos los PLC es una tarea que difícilmente podía ser completada, puesto que cada fabricante tenía herramientas y protocolos que solo trabajaban bajo sus estándares y dispositivos, por lo

cual en la actualidad hay 5 lenguajes estandarizados de PLC a nivel mundial, los cuales difieren grandemente entre sí.

Es por esto que el software libre también tocó a las puertas de la automatización, la aparición de tecnologías embebidas como arduino en el año 2005 y Raspberry en el 2009, ha cimentado el comienzo de una revolución en el campo de la automatización, no solo en procesos académicos y de aprendizaje, sino también en campos de aplicación en diferentes sectores de la economía. Sin embargo, estos dispositivos embebidos por sí solos pueden llegar a ser muy inestables en un entorno industrial debido al gran ruido al que se exponen los equipos en un ambiente con grandes motores y máquinas que demandan gran cantidad de energía. Es esta la razón por la que empresas como KUNBUS, Industrino, Controlino, Industrial Shields y otras en el mercado, han decidido dar un paso adelante diseñando y construyendo dispositivos de control que prometen fiabilidad para manejar maquinaria de carácter industrial, con placas programables bajo la flexibilidad y economía del software libre.

Ciclo de programa de un PLC

Un PLC funciona como un PC, con la diferencia que está optimizado para su uso industrial, y en vez de ratón y teclado tiene periféricos de entrada y salida. Tiene un procesador que recoge los datos de los sensores y toma una foto al sistema, el cual se denomina Escaneo de imagen, y captura la información que hay presente en las entradas y las salidas en ese instante. Teniendo en cuenta que el Ciclo de escaneo de un PLC es el tiempo que le toma al PLC capturar, transportar y procesar la imagen de los datos, para estar listo y capturar la próxima, a medida que se necesite mayor información de entradas y / o salidas, se debe pensar en dispositivos con más KHz de procesamiento.

Kunbus

Entre las más destacadas tenemos a Kunbus, uno de sus nichos principales de mercado es la automatización industrial basada en la programación de software libre disponible para Raspberry, para lo cual ofrecen una variedad de equipos que cumplen diferentes funciones, núcleo o cerebro, módulo base, y compuertas de conexión a diferentes protocolos de redes, que se pueden integrar como un solo sistema robusto y programable, su línea de PLC's, conocida como "Revolution PI" , los cuales son:

1. **CORE (núcleo-servidores):** Esta empresa ofrece 3 tipos de dispositivo principal, con la inclusión de un nuevo integrante de la familia mientras se desarrollaba este proyecto, Rev Pi Connect.

La gran diferencia entre ellos es la capacidad de cómputo, el Rev Pi core es un dispositivo diseñado para gestionar procesos y tareas sencillas, con solo un núcleo de procesamiento; mientras que RevPi Core 3 está equipado con la versión 3 del módulo de cómputo de Raspberry PI, y un procesador de 4 núcleos que permite desarrollar tareas que requieren alto nivel de cómputo tales como el procesamiento de imágenes, sin embargo los costos no difieren mucho, como se puede apreciar a continuación en la *Tabla 1, Comparación de características especiales de los 3 tipos de módulos PLCs Core de la empresa Kunbus.*

Tabla 1

Comparación de características especiales de los 3 tipos de módulos PLCs Core de la empresa Kunbus

	RevPi Core	Rev Pi Core 3	RevPi Connect
Número de núcleos	1	4	4
Procesador Clock rate	Broadcom BCM2835 - 700 Mhz	Broadcom BCM2837 - 1.2 Ghz	Broadcom BCM2837 quad-core ARM Cortex A53 (ARMv8)
RAM	500Mb	1Gb	1GB
Sistema	Raspbian Wheezy (Parche RT	Raspbian Jessie (Parche RT	Raspbian Jessie (Parche RT

Operativo	incluido)	incluido)	incluido)
Precio Euro / Peso	€211 (\$ 749.521 Aprox)	€235.62 (\$ 836.977 Aprox)	€332.01 (\$1'179.342 Approx)

Fuente: <https://www.kunbus.com> (Consulta 17 octubre 2018)

Adaptado de: Industrial ethernet and fieldbus solution from KUNBUS

Como se mencionó anteriormente, se ha incluido RevPi Connect, el cual es un dispositivo diseñado especialmente para el Internet de las cosas, consta de 4 núcleos para procesamiento de alto rendimiento, y le da al usuario gran libertad debido a su concepto de plataforma abierta que incluye todos los derechos “Root” de programación en python en un SO Raspbian optimizado para IoT, añadido a la opción de conectar gran cantidad de periféricos de entrada y salidas, con la posibilidad de conexión a 2 redes ethernet, una para control y monitoreo del sistema y por otro lado la recepción, captura y posible subida de datos a un servidor en la nube.

- 2. PLCs módulo base** En este segmento, la empresa ofrece 3 opciones diferentes variando un poco el número de entradas y salidas según el proyecto requiera, inclusive, seleccionando solo entradas o solo salidas, de acuerdo a la comparación que se puede apreciar en próximas líneas en *Tabla 2, Comparación de las características principales de los módulos de E / S disponibles de los PLCs módulo base.*

La gran diferencia entre ellos es la capacidad de cómputo, el Rev Pi core es un dispositivo diseñado para gestionar procesos y tareas sencillas, con solo un núcleo de procesamiento; mientras que RevPi Core 3 está equipado con la versión 3 del modulo de computo de Raspberry PI, y un procesador de 4 núcleos que permite desarrollar tareas que requieren alto nivel de computo tales como el procesamiento de imágenes.

Tabla 2

Comparación de las características principales de los módulos de E / S disponibles de los PLCs módulo base

		Entrada	Salida	Precio
PLC I/O Digitales	RevPi DIO	14 digitales 24 V / 2.4mA	14 digitales hasta 500mA / contrafase 100mA	€177.31 / \$625.920

	RevPi DI	16 digitales 24 V / 2.4mA	--No tiene--	€141.61 / \$499.895
	Rev Pi DO	--No tiene--	14 digitales 500mA / contrafase 100mA	€153.51 / \$541.904
PLC I/O Análogos	RevPi AIO	4 analogicas 24 bit convertidor a AD	2 analogicas 16 bit - A / V	€332.01 / \$1'171.989

Fuente: <https://www.kunbus.com> (Consulta 17 octubre 2018)

Adaptado de: Industrial ethernet and fieldbus solution from KUNBUS

Todas las versiones tienen separación galvánica entre el circuito lógico PiBridge y las entradas y salidas. La versión RevPi DIO tiene además separación galvánica entre entradas y salidas. Todas las versiones están protegidas contra interferencias según los requisitos EN61131-2.

También se ofrece un dispositivo de expansión, con 4 entradas y 2 salidas análogas y 2 canales RTD. RevPi AIO cuenta con un conector de E / S de 28 pines en la parte delantera. Las entradas, salidas y canales RTD están aislados galvánicamente entre sí y con el circuito lógico. RevPi AIO está protegido contra interferencias de acuerdo con los requisitos EN61131-2.

3. Gateway : Debido a que no es fácil integrar un dispositivo en una red industrial, y menos un dispositivo embebido, este es tal vez el punto de mayor atención para el portafolio de esta empresa. Protocolos especiales se utilizan a menudo para la transmisión de datos, como Profinet o Profibus. La empresa afirma que el uso de sus GATEWAY, llamadas RevPi Gate, integran Revolution Pi en una red industrial sin problema, sin importar el protocolo al que se quiera acceder. La única recomendación para utilizar alguna de estas compuertas, es usar Servidores de 4 núcleos. Son bastantes las opciones y los precios son accesibles, seguido, *Tabla 3, Gateways para conexión a otras redes industriales bajo los más conocidos protocolos* lo ilustra.

Tabla 3

Gateways para conexión a otras redes industriales bajo los más conocidos protocolos

Protocolo	Maestro	Esclavo	Bus de datos	Ancho de banda máx	Datos IN / OUT	Precio
PROFINET IRT			2 x RJ45	100 Mbit/s	512 Bytes IN/OUT	€214.207/ \$774.501
Ethernet / IP			2 x RJ45	100 Mbit/s	480 Bytes IN/OUT	€166.60/ \$600.800
POWERLINK			2 x RJ45	100 Mbit/s	512 Bytes IN/OUT	€246.33/ \$890.400
EtherCAT			2 x RJ45	100 Mbit/s	512 Bytes IN/OUT	€161.84/ \$583.000
Sercos III			2 x RJ45	100 Mbit/s	512 Bytes IN/OUT	€246.33/ \$890.400
Modbus TCP			2 x RJ45	10 / 100 Mbit/s	480 Bytes IN/OUT	€166.60/ \$600.800

PROFIBUS			D-SUB 9F	up to 12 Mbit/s	244 Bytes IN/OUT	€201.11/ \$727.500
DeviceNet			5 pole / pitch 5.08	up to 500 Kbit/s	512 Bytes IN/OUT	€148.75/ \$535.700
CANopen			D-SUB 9M	up to 1 Mbit/s	512 Bytes IN/OUT	€149.94/ \$509.400
Modbus RTU			8 pole / pitch 3.50	up to 115.2 Kbit/s	512 Bytes IN/OUT	€165.41/ \$597.200
Serial			8 pole / pitch 3.50	up to 115.2 Kbit/s	512 Bytes IN/OUT	€165.41/ \$597.200
DMX			8 pole / pitch 3.50	-	512 Bytes IN/OUT	€165.41/ \$597.200

Fuente: <https://www.kunbus.com> (Consulta 17 octubre 2018)

Adaptado de: Industrial ethernet and fieldbus solution from KUNBUS

Industrial Shields

Es una empresa que a pesar de ser relativamente nueva, con tan solo 6 años después de su fundación, *(en comparación de gigantes de la industria de la automatización como Allen Bradley, cuya trayectoria se remonta a la década del 70)* ya se ha extendido por más de 40 países a través de una red de distribuidores e integradores que se han asociado a lo largo de los años. Su rápida expansión a nivel mundial también ha sido posible debido al gran impacto que tienen las ventas online en su página de internet, de fácil navegación y constante desarrollo, medio por el cual fue descubierta esta empresa, que se convertirá en el foco de esta investigación.

Como podemos observar en *Tabla 4, Comparación de las diferentes tecnologías de controladores actualmente disponibles*, los directivos de Industrial Shields aseguran que sus equipos incorporan los beneficios de un controlador Arduino, y los mezclan con la robustez que requieren los PLC industriales, de esta manera toma mucha fuerza la programación a través de software libre, que puede significar una gran reducción de costos, también la variedad de protocolos de comunicación como el USB, Bluetooth, Ethernet, entre otros, llevada a un entorno más allá del académico, viable para un uso directo industrial, bajo la norma internacional de trabajar con actuadores y sensores a 24 V; por lo que al ser comparados con las demás tecnologías actuales para el control y automatismo industrial, toman gran ventaja mezclando lo mejor de ambos mundos.

Tabla 4

Comparación de las diferentes tecnologías de controladores actualmente disponibles

PLC	Arduino	Industrial Shields	Panel PC	Raspberry
Programación basada en un estándar	Programación directa de uC (Processing)	Programación directa de uC (Processing)	Windows	Raspbian (Linux)
Software cerrado	Software libre	Software libre	Software cerrado	Software libre
Sistemas de comunicación habituales, protocolos cerrados	Comunicación Rx, Tx, (TTL 5V), USB, I2C, SPI pueden necesitar adaptación	RS 485, RS 232, I2C, SPI, USB, Serial TTL, Ethernet	USB, Ethernet, I2C...	USB, Ethernet, I2C...
I/Os predefinidas	I/Os configurables	I/Os configurables (analog, digital)	I/Os predefinidas	GPIOs configurables
24 Vcc NPN, PNP, etc	5Vcc, Pull up, Pull down, etc	24 Vcc PNP. Relay	24 Vcc, PNP, NPN, etc	5 Vcc, Pull up, Pull down, etc
Uso directo industrial	Prototipado / Formación / DIY	Uso directo industrial	Uso directo industrial	Prototipado / Formación / DIY
Carril DIN	PCB	Carril DIN	Carril DIN	PCB

Fuente: (Consultado 1 Octubre 2018)

(<http://www.automaticaeinstrumentacion.com/es/downloads2/industrial-shields-albert-prieto.compressed.pdf>)

Tomado de: *Presentación modelo propio industria 4.0, Albert prieto, 2016*

Esta empresa tiene un amplio portafolio de productos y servicios, entre los que se encuentran sensores de corriente y temperatura, placas o CPU basadas en Arduino, fuentes de poder y muchos mas accesorios para los dispositivos PLC principales. También se destaca la oferta de hardware para el prototipado industrial de proyectos de IoT, usando una placa llamada OpenMote B y una antena para interiores, basados en un software open source llamado Contiki OS. Sin embargo, el pilar de esta empresa ha sido el desarrollo de PLCs para automatización industrial basada en Software Open Source, con los que se posicionaron en el mercado a nivel mundial, incluso hasta 2016, solo se ofrecían los dispositivos para este nicho de mercado. Hay una gran variedad de dispositivos que son ofrecidos actualmente por esta empresa para la rama de la automatización, y se dividen en 2 categorías, PLCs y Panel PCs.

- 1. PLCs:** La mayoría de todos los PLCs comparten una similitud de acuerdo al tipo de dispositivo que se va a utilizar, como lo son entradas listas para ser usadas por tensión de alimentación de 5, 12 y 24 Voltios, y la posibilidad de configurar el tipo de entrada según sean las necesidades del proyecto, con un dispositivo con la capacidad máxima de hasta 58 Entradas / Salidas, protocolos de comunicación que incluyen USB, puerto serial, modbus, RS232, RS485, I2C, SPI y comunicación para codificación, envío / recepción de datos y múltiples módulos y conexiones de sensores.

Además del antes mencionado gratuito y en constante desarrollo lenguaje de programación Arduino, en su plataforma Arduino IDE, ya que estos dispositivos cuentan con placas originales Arduinos que son incluidas y adaptadas para una gama de automatización e instrumentación industrial.

- a. 20 I/Os PLC Ardbox:** Como su nombre lo indica, son dispositivos que tienen un número límite de 10 entradas, configuradas como analógicas o discretas y 10 salidas, según se requiera, analógicas, discretas o inclusive por relé, de acuerdo con *Tabla 5, Comparativa de los dos tipos de PLC all in de esta rama, su única diferencia es la salida*, para un total de 20 I/O; se considera para trabajar en proyectos que no exigen gran cantidad de entradas o salidas y que tampoco requieren una alta complejidad de cómputo.

Tabla 5

Comparativa de los dos tipos de PLC all in de esta rama, su única diferencia es la salida

Nombre dispositivo	PLC Arduino ARDBOX 20 I/O Analog HF Modbus	PLC Arduino ARDBOX PLC 20 I/Os RELAY HF
Entradas	10 Entradas (5 to 24Vdc): Analógicas (0-10Vdc) y Digitales (5 to 24Vdc)	10 Entradas (5 to 24Vdc): Analógicas (0-10Vdc) y Digitales (5 to 24Vdc)
Salidas	10 Salidas (5 to 24Vdc): Analógicas (0-10Vdc) y Digitales/PWM Optoaisladas (5 to 24Vdc)	8 Salidas: Relé 220Vac
Comunicación	USB, RS485 Half-Full Duplex (Modbus), I2C, SPI, RS-232	USB, RS485 (Half-Full Duplex), I2C, SPI, MODBUS, RTU disponible, RS-232
Precio	135,00 € (\$489.900 Aprox)	140,00 € (\$508.100 Aprox)

Fuente: https://www.industrialshields.com/es_ES/slides-product-range-arduino-industrial-plc

(Consultado 10 de octubre 2018)

Adaptado de: Arduino based PLC controller for Industrial Applications by Industrial Shields

b. Ethernet PLC M-duino:

Son dispositivos más robustos, que tienen un número más grande de entradas y salidas y soportan una cantidad mayor de protocolos de comunicación, entre los que más se destaca Ethernet, el cual es el protocolo que ha tomado mayor importancia en los últimos tiempos para transmisión de datos a alta velocidad y remotamente. Veamos una comparación en detalle en *Tabla 6 Comparación de dispositivos PLC con conexión a Ethernet que tienen la capacidad de comunicarse a través de Ethernet* en donde el dispositivo que está

seleccionado en la tabla es el que más adelante se seleccionará como referencia.

Tabla 6,

Comparación de dispositivos PLC con conexión a Ethernet que tienen la capacidad de comunicarse a través de Ethernet

Nombre dispositivo	In	Out	Comunicación	Precio
M-DUINO PLC Aduino 57AAR I/Os Analog / Digital / Relay PLUS	32 Entradas	25 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	355,00 €/ \$1'284.900
M-DUINO PLC Arduino Ethernet 19R I/Os Relé / Analog / Digital PLUS	6 Entradas	11 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	269,00 €/ \$973.600
M-DUINO PLC Arduino Ethernet 21 I/Os Analog/Digital PLUS	13 Entradas	8 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	196,00 €/ \$709.400
M-DUINO PLC Arduino Ethernet 38AR I/OsAnalogi / Digital / Relé PLUS	19 Entradas	19 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	312,00 €/ \$1'129.200
M-DUINO PLC Arduino Ethernet 38R I/Os Relé / Analógico / Digital PLUS	12 Entradas	22 Salidas	Ethernet, USB,RS485, RS232, I2C, Modbus	349,00 €/ \$1'263.100
M-DUINO PLC Arduino Ethernet 42E/S Analógicas/Digitales PLUS	26 Entradas	16 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	275,00 €/ \$995.300
M-DUINO PLC Arduino Ethernet 50RRA E/S Relé / Analógicas / Digitales PLUS	18 Entradas	31 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	383,00 €/ \$1'386.200
M-DUINO PLC Arduino Ethernet 53ARR E/S Relé / Analógicas / Digitales PLUS	25 Entradas	28 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	383,00 €/ \$1'386.200
M-DUINO PLC Arduino Ethernet 54ARA E/S Analógicas/Digitales PLUS	29 Entradas	25 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	355,00 €/ \$1'284.903
M-DUINO PLC Arduino Ethernet 57R E/S Relé / Analógicas / Digitales PLUS	18 Entradas	31 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	420,00 €/ \$1'520.100
M-DUINO PLC Arduino Ethernet 58 E/S Analógicas/Digitales PLUS	36 Entradas	22 Salidas	RTC, µSD, Ethernet, USB, Full/Half Duplex RS485, RS232, I2C, Modbus	320,00 €/ \$1'158.200

Fuente: https://www.industrialshields.com/es_ES/slides-product-range-arduino-industrial-plc

(Consultado octubre 17)

Adaptado de: Arduino based PLC controller for Industrial Applications by Industrial Shields

2. Panel PC:

Son pantallas táctiles resistivas, todas de 10.1" a base de placas programables en software libre que permiten la visualización, monitoreo e incluso programación de manera visual de datos, gráficas, informes, etc en tiempo real. Se dividen principalmente por el tipo de tarjeta de control que se quiera trabajar según las necesidades del proyecto, como se permite apreciar en la próxima *Tabla 7*, *Comparativa de los cuatro tipos de dispositivos de esta rama HMI o Panel PC.*

Tabla 7

Tabla comparativa de los cuatro tipos de dispositivos de esta rama HMI o Panel PC

Nombre dispositivo	BananaTouch Pi 10.1"	HummTouch Pi 10.1"	TouchBerry PI 10.1"	Panel PC Industrial EMC Aluminio (3 Opciones de placa)
Placa base	Banana Pi	HummingBoard	Raspberry PI	-Raspberry PI -Banana PI -Hummingboard
Sistemas operativos que soporta	SO Linux, Android o Windows 10 IoT.	SO Linux o Android.	SO Linux o Windows 10 IoT.	Según la placa base
Precio	525,00 € (\$1,907,037 Approx)	563,00 € (\$2,045,257 Approx)	495,00 € (\$1,798,043 Approx)	425,00 € (\$1,543,800 Approx)

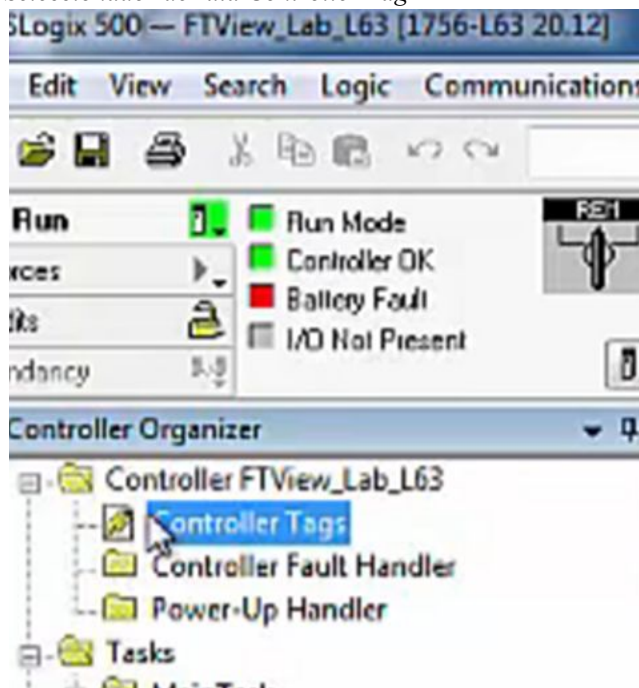
Fuente: https://www.industrialshields.com/es_ES/slides-product-range-arduino-industrial-plc (Consulta 18 de Octubre 2018)

Adaptado de: Arduino based PLC controller for Industrial Applications by Industrial Shields

Configurar un tópico OPC/DDE en RsLinx para obtener datos de un controlador Logix 500 y agregarlo a una hoja de cálculo de excel

Se define tópico como una selección de variables o arreglos de datos que contienen históricos o información de sensores en tiempo real, los cuales se quieren transmitir hacia un programa fuera de la suite de Rockwell simulando un túnel de datos. Lo primero que se hace para configurar un tópico, es abrir la aplicación del Rslogix 500, en la parte izquierda de la pantalla seleccionar la ubicación de los datos, en este caso es “Controller tags”, bajo la ventana *controller organizer*, como lo muestra la *Figura A1, Seleccionador de ruta Controller Tag*.

Figura A1
Seleccionador de ruta Controller Tag



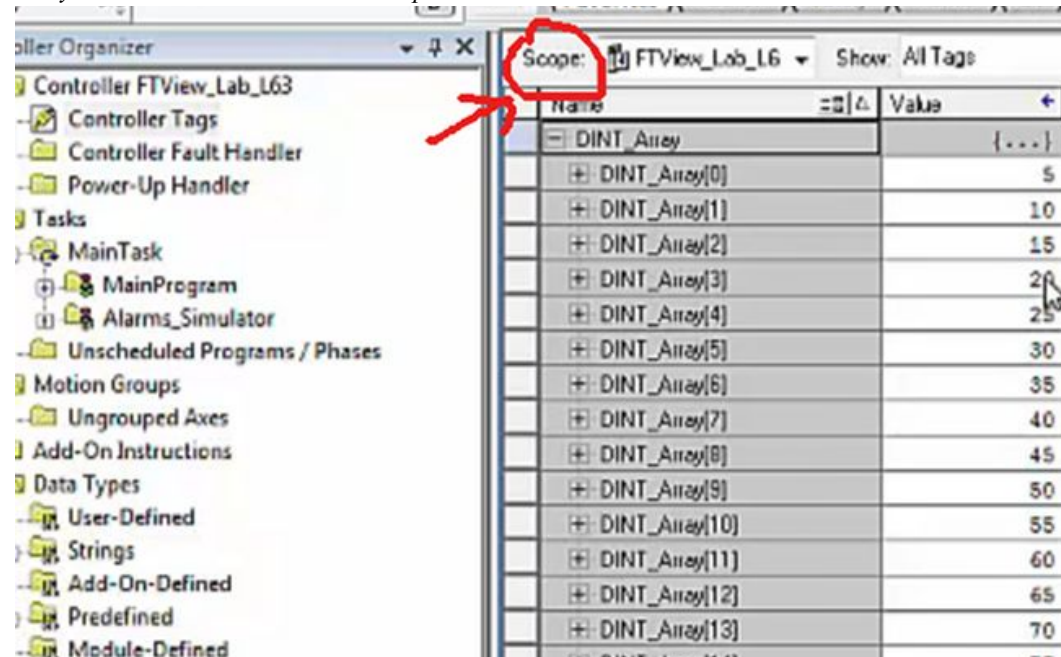
Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

Para mostrar la transmisión de datos históricos contenidos en un array, en este ejemplo se le llama DINT-Array[], son valores aleatorios que pueden ver en a continuación en *Figura A2, Array DINT contenedor de datos a exportar*, posteriormente cambiando el

scope, remarcado con rojo, al dar click en la pestaña contenedora del scope datos a exportar, se puede ver una variable *temperature* bajo un Scope llamado *Alarms_Simulator* de la cual también se van a exportar los datos desde el PLC Allen Bradley hacia Excel, como se puede apreciar más abajo en *Figura A3, Variable temperatura que toma datos cambiantes provenientes de un sensor de calor.*

Figura A2

Array DINT contenedor de datos a exportar



Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

Figura A3

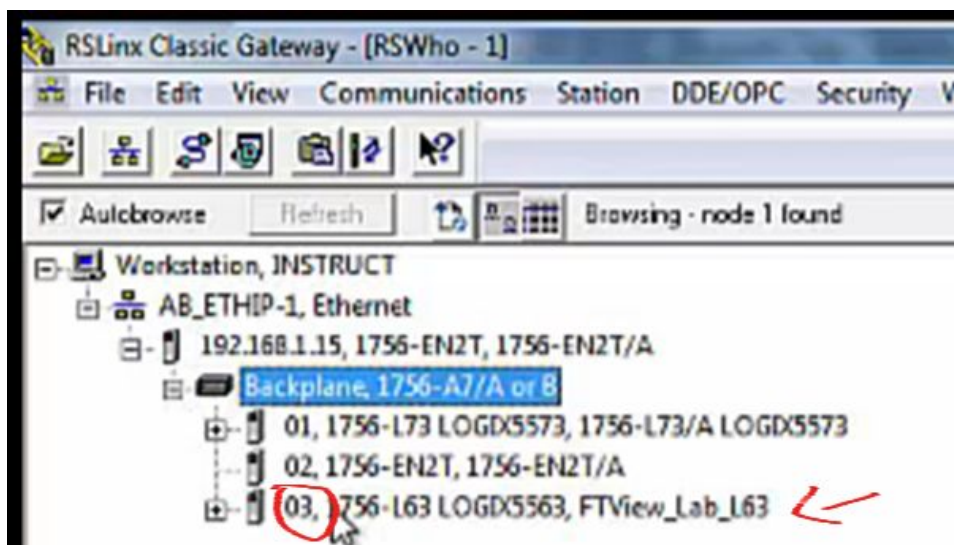
Variable temperatura que toma datos cambiantes provenientes de un sensor de calor

Scope: Alarms_Simulator		Show: All Tags
Name	Value	Format
+ Current_Segment	1	
Decrease_Temp	0	
Increase_Temp	1	
+ Input_Value	50	
+ One_Shot	{...}	
+ Ramp_Value	{...}	
+ RMPS_01	{...}	
+ Sim_Time	{...}	
+ Soak_Time	{...}	
+ Soak_Value	{...}	
+ Temperature	407	
Temperature1	200.1	

Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

Se procede a ingresar al programa RSLinx, el cual debe ser una versión que soporte la comunicación DDE OPC, no puede ser una versión gratuita o Lite, pues para el proyecto se supone que las empresas que usan este software lo usan completamente licenciado, el cual permite la conexión de uno o varios PLC a una estación de trabajo, para este ejemplo podemos ver que ya se tiene un PLC Micrologix 1100 conectado a través de ethernet a una versión RSLinx Classic Gateway, en la parte izquierda de la *Figura A4 Seleccionador del controlador número 3, el cual contiene los datos a exportar*, expandiendo la pestaña “Workstation, INSTRUCT” podemos ver que solo hay un PLC conectado “AB_ETHIP-1, Ethernet”, bajo el una dirección IP configurada, que corresponde a ese equipo y que contiene varios drivers de conexión. El controlador que contiene los datos que para este ejemplo están variando es el controlador “03, 1756-L63 LOGIX 5563, FTVIEW_lab_L63”, que se muestra señalado en rojo y con el puntero en la *Figura A4* a continuación.

Figura A4
Seleccionador del controlador número 3, el cual contiene los datos a exportar

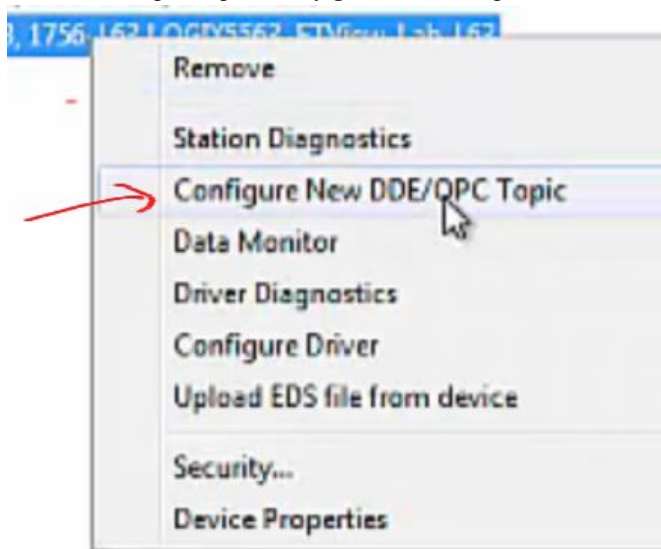


Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
 Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

El paso a seguir es dar click derecho en el controlador que contiene los datos que queremos capturar y seleccionar “Configure New DDE/OPC Topic”, como se puede ver en *Figura A5 Selección de opción para configuración del tópico DDE/OPC.*

Figura A5

Selección de opción para configuración del tópico DDE/OPC

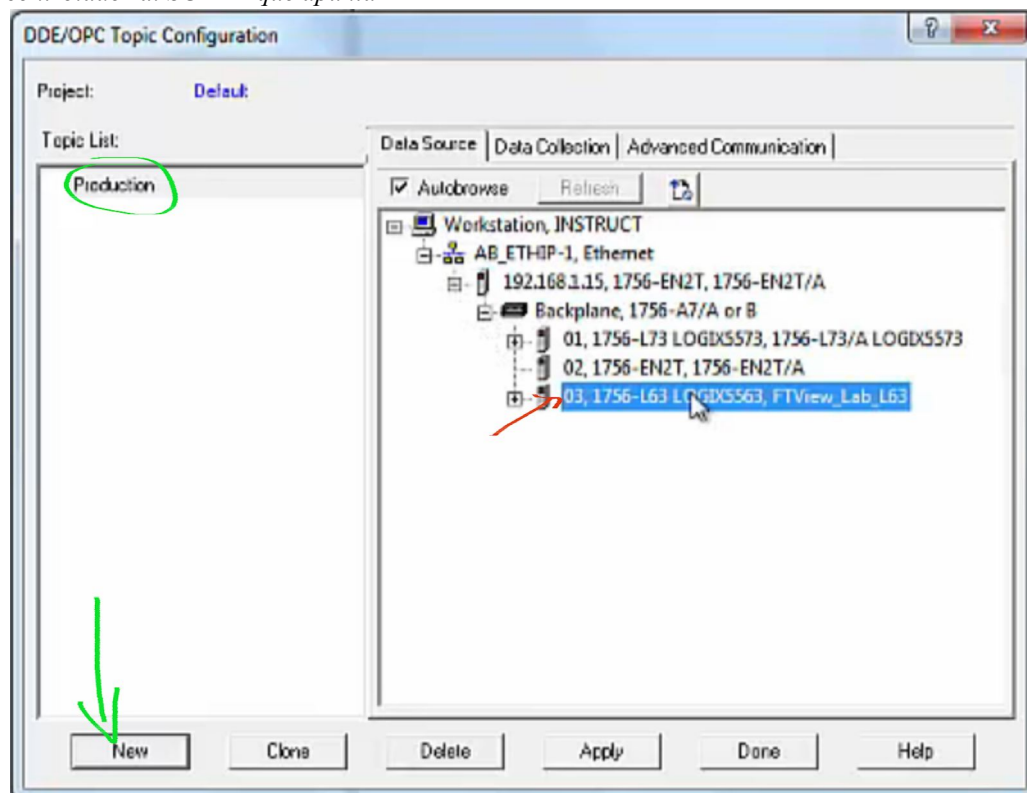


Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
 Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

También es posible acceder desde la parte superior en la pestaña de “DDE/OPC” la primer opción que es “Topic Configuration”, cualquiera de estas dos opciones abre la misma ventana, en la cual se puede ver la lista de tópicos en la parte izquierda de la pantalla. Si es la primera vez que se va a configurar, se debe dar click en el botón “New”, visible en parte inferior derecha de la siguiente *Figura A6, Configuración de primer tópico llamado Production* lado izquierdo de la figura, lado derecho controlador al SCADA que apunta. A este nuevo tópico se le ha llamado “Production”, y es puesto a apuntar al controlador número 3, de donde se escogieron los datos en los primeros pasos.

Figura A6

Configuración de primer tópico llamado Production lado izquierdo de la figura, lado derecho controlador al SCADA que apunta



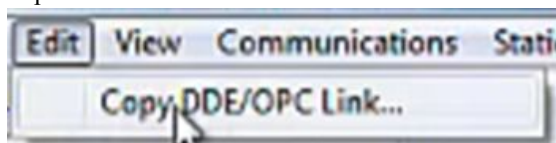
Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

En la izquierda se aprecia el nombre del nuevo tópico y a la derecha el controlador seleccionado, al dar click en el botón “Apply” que está en la parte inferior central de la anterior Figura A6, hemos finalizado la tarea de crear el tópico, que funciona como túnel

cerrado de comunicaciones y transmitirá las variaciones bidireccionales de datos con una hoja de cálculo, de esta manera ya es posible exportar estos datos para su procesamiento en otra aplicación, en este caso, Microsoft excel.

El próximo paso es copiar el link que va a permitir comunicación entre el PLC y la hoja de cálculo, por lo que se procede a seleccionar en el menú superior izquierda de RSLinx, como se puede apreciar en *Figura A7, copia del link DDE/OPC en RSLinx*.

Figura A7
Copia del link DDE/OPC en RSLinx

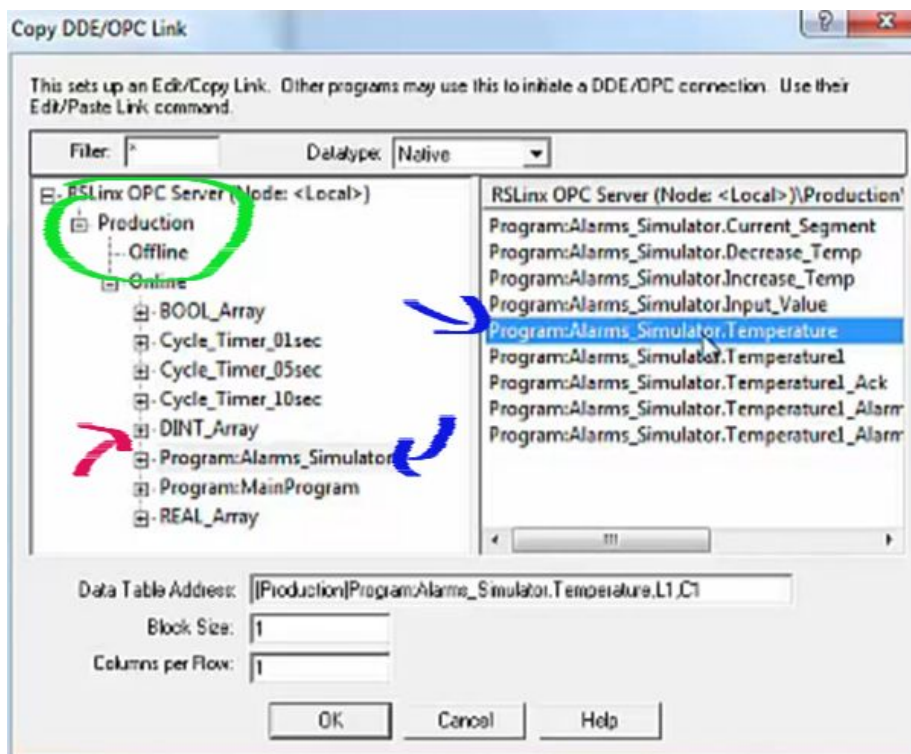


Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

En la pestaña “edit”, la única opción “Copy DDE/OPC link...” despliega una ventana emergente para escoger la información específica a recolectar para ser exportada como se puede ver más adelante en *Figura A8*.

Una vez abierto el cuadro de interacción, en la parte izquierda superior bajo la pestaña producción, señalada en la figura A8 en un círculo verde, se despliega su línea de programas y al dar click “Alarms_Simulator”, quien contiene en su programación la variable temperatura, debe desplegarse en la parte derecha del recuadro, todas las variables de este programa. como se aprecia en la *Figura A8, Selección de la variable “temperature” para inicializar la conexión*, La variable exacta que ha sido señalada con flechas azules, es la misma que apreciamos anteriormente.

Figura A8
Selección de la variable “temperature” para inicializar la conexión



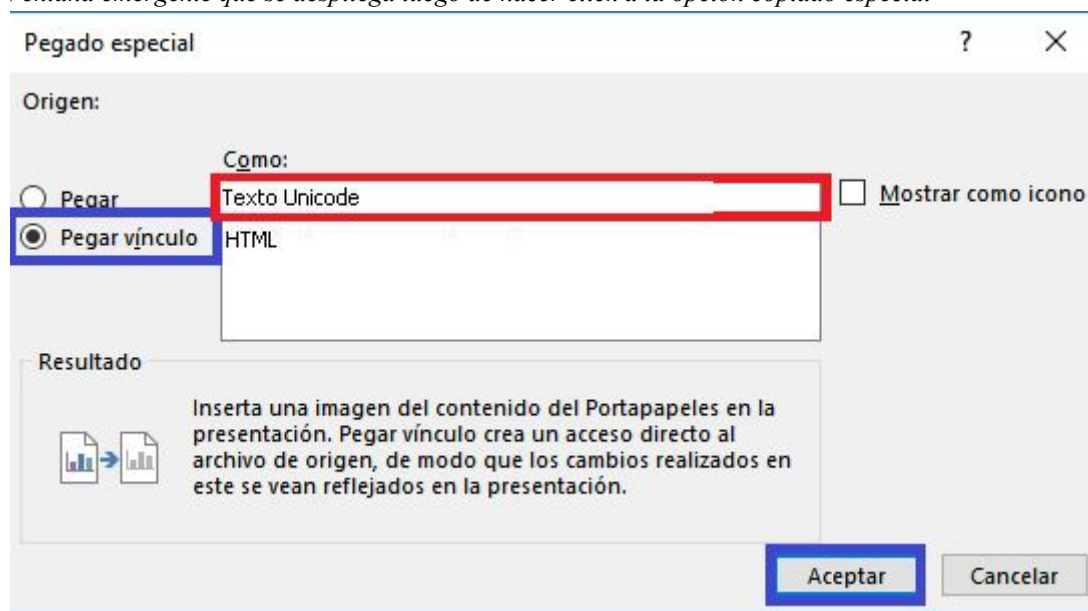
Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
 Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

El paso siguiente es abrir de la suite de office, el programa Microsoft Excel, desde su versión 2007 compatible con los protocolos mencionados; cabe decir que entre más actual sea la versión de office, mejor será la compatibilidad con este OPC. Una vez abierta una nueva hoja de cálculo, en alguna de las celdas, para este ejemplo E5, se presiona clic derecho, opción *pegado especial*, esto abre una ventana emergente, como se aprecia a continuación en la *Figura A9, Ventana emergente que se despliega luego de hacer click a la opción copiado especial*. Es importante tener en cuenta que solo la opción de pegado especial permitirá la transmisión de datos con esta hoja de calculo, posterior a ello, en la parte izquierda de la ventana emergente, se cambia a la opción “Pegar vínculo”, visto en la próxima figura en el recuadro azul, este crea un enlace a el archivo fuente, seleccionando “*Texto unicode*” sugerido con el recuadro en rojo. Este vínculo que hace parte de la suite de Rockwell se incrusta en Microsoft Excel, y es aquí donde las librerías y protocolos OPC y OLE desarrolladas por la empresa desarrolladora del Sistema Operativo Windows se comunica con la suite de rockwell, permitiendo que todos los datos históricos seleccionados en el túnel OPC

y los cambios que sufra la variable temperatura sean actualizados y reflejados en la hoja de cálculo en cuestión de centésimas de segundo.

Figura A9

Ventana emergente que se despliega luego de hacer click a la opción copiado especial

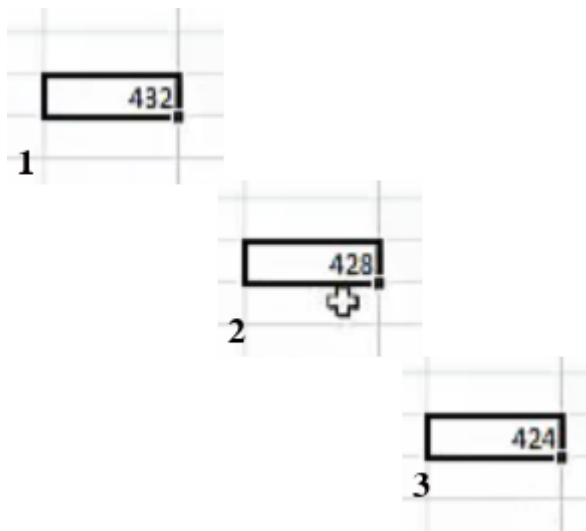


Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

El resultado en la hoja de excel es apreciable en seguidas líneas en Figura A10, Secuencia izquierda a derecha de capturas de pantalla en la celda E5 hecha con 3 segundos de retraso entre sí. Este refleja la variación en tiempo real que sufre el valor de la celda en donde fue pegado el link, en este caso temperatura.

Figura A10

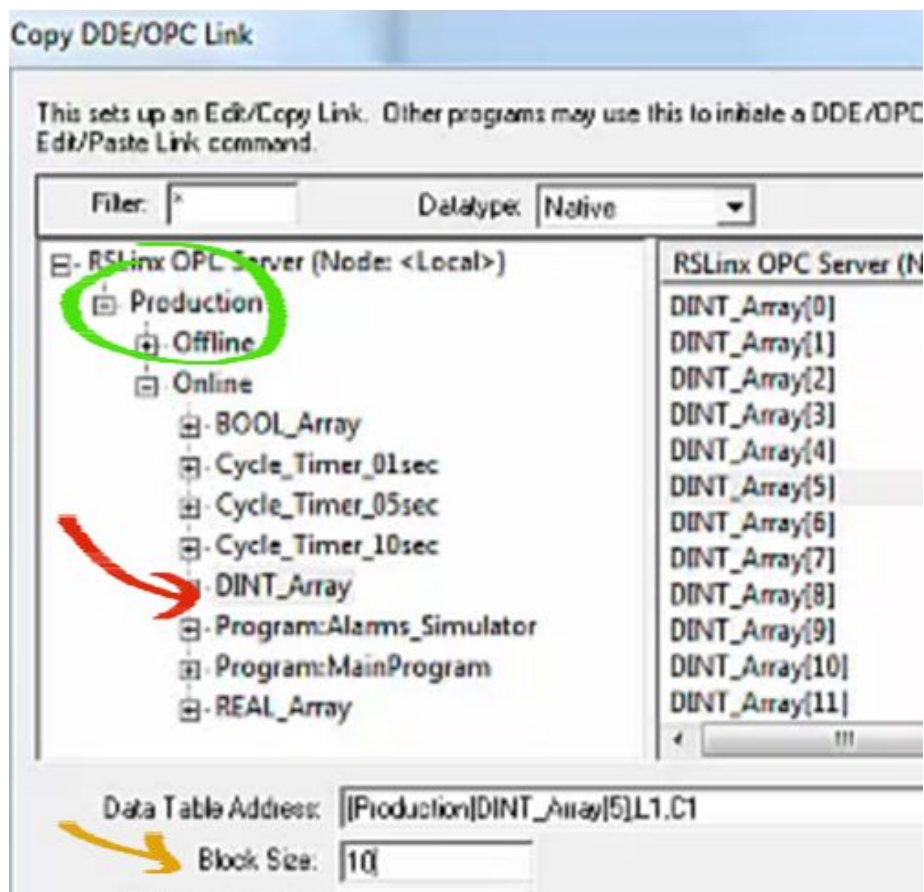
Secuencia de capturas de pantalla en la celda E5 hecha con 3 segundos de retraso entre sí



Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
 Adaptado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

En la figura anterior se ve la manera en cómo se conecta y se actualizan los datos en tiempo real, puesto que se está copiando la información directamente de la variable temperatura, y con un tamaño de bloque 1, información en tiempo real en una sola celda. Nuestro objetivo es poder realizar históricos, debido que estos son los datos que interesan en la gerencia de las compañías, por lo que se hace necesario el llamado del arreglo de datos previamente invocado DINT_Array. El proceso de copiado de link es el mismo con ciertas diferencias visibles en *Figura A 11, Selección de los datos del array DINT_Array[]*

Figura A11
 Selección de los datos del array DINT_Array[] para inicializar la conexión



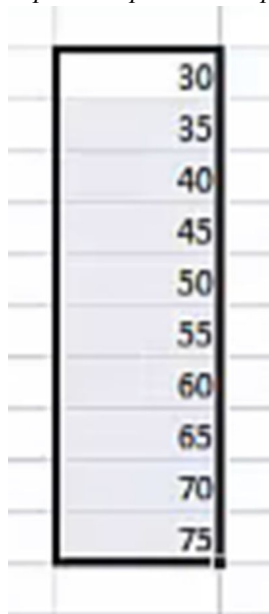
Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]
 Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

En esta ventana emergente debe seleccionarse en la parte superior izquierda, el t pico creado, para este caso “Production”, sealado en verde, el cual despliega un men  que contiene las variables y programas que son elegibles, como ya se hab a mencionado antes, el arreglo DINT_Array[] contiene un serie de datos hist ricos, y es seleccionado como se puede ver en la figura A 11 sealado por la flecha roja. Por  ltimo se aprecia una flecha naranja en “Block size” (Tama o de bloque), se ingresa el numero 10 para indicar que el tama o de datos que se va a exportar resulta en una tabla de 10 x 1 en excel, una vez seleccionados los par metros se procede a dar click en el bot n “Ok”.

A continuación se repite el paso de pegado especial, visto anteriormente en la figura A 9, esta vez en la celda H5, y el resultado es visto en la siguiente *Figura A12, Captura de pantalla después de pegado especial de datos del array en bloque 1 x 10*:

Figura A12

Captura de pantalla después de pegado especial de datos del array en bloque 1 x 10



Fuente: <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s> [Consultado el 5 de Octubre 2018]

Tomado de: Youtube, canal de Brett Allen, Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel

Este es el proceso que ilustra los pasos necesarios en orden de obtener los datos del Micrologix 1100 a través de RSLinx que hace parte de la suite de Rockwell Automation, siendo un OPC, y permitiendo el flujo de datos desde el dispositivo cerrado, a un archivo en excel desde donde estos datos pueden ser exportados a una base de datos local, o a través de un servidor HTML que efectúe peticiones con cierto intervalo de tiempo a la(s) celda(s) que contienen la información que desea ser compartida entre PLCs.

Referencias

Allen, Brett. (2016). Video de Youtube: *Configure RSLinx DDE OPC Topic to Get Data From a Logix Controller to MS Excel*. Disponible en <https://www.youtube.com/watch?v=uSqsJApBGDs&t=1s>

ARDUINO. (s.f). *¿ Qué es arduino?*. Recuperado de <https://arduino.cl/que-es-arduino/>

Automation Networks. (s.f). *RSLinx Classic and enterprise software*, Recuperado de: <http://automation-networks.es/glossary/rslnx-classic-and-enterprise-software>

CTIN. Centro de Tecnología e Innovación. (2013). *Que es un PLC ?*. Ciudad de Mexico, recuperado de <http://www.ctinmx.com/que-es-un-plc/>

GL Ingenieros. (s.f). *Automatización y Calidad de la Potencia*. Recuperado de <http://www.glingenieros.com.co/calidad-de-energia/>

Industrial Shields. (s.f). *Quienes somos, Controlador PLC Industrial basado en Arduino, de Industrial Shields*. [Consulta 30 agosto 2018], Recuperado de: https://www.industrialshields.com/es_ES/about-us

Industrial Shields. (2016). *Manual de Familia M-Duino (Págs 10-16)*. [Consulta 28 de Octubre 2018]. Recuperado de: https://www.plcarduino.com/wp-content/uploads/2017/05/23.03.16_M-duino-Family-PLC.pdf

Ing. Cesar Augusto. (Usuario Steemit: @autinf). (2018). *La pirámide de automatización*. Recuperado de: <https://steemit.com/spanish/@autinf/la-piramide-de-automatizacion>

International Standard. (2007). *IEC 61131-2 Tercera edición, Programmable controllers – Part 2: Equipment requirements and tests*. Recuperado de https://webstore.iec.ch/p-preview/info_iec61131-2%7Bed3.0%7Den.pdf

KUNBUS. (s.f). *Meet the revolution PI products - Industrial Raspberry PI*. (Consulta 20 de octubre 2018). Recuperado de <https://revolution.kunbus.com/revolution-pi-series/>

KUNBUS. (s.f). *Meet the revolution RevPi Connect base module - Industrial Raspberry PI, products*. Consulta 20 de octubre 2018. Recuperado de <https://revolution.kunbus.com/revpi-connect/>

Marsellach, Francesc. FMJ Ingenieros (2012). *Introducción a la automatización*. Recuperado de <https://www.fmjingenieros.com/servicios/automatizacion-industrial/introduccion-a-la-automatizacion>

Open Source Organization. (s.f). *The Open Source Definition | Open Source Initiative*. Recuperado de <http://opensource.org>

Openmote. (s.f) Open Hardware for the industry of internet of things. [Consulta 18 noviembre 2018], Recuperado de <http://www.openmote.com/>

OPC Foundation. (2018). *What is OPC?*. Recuperado de: <https://opcfoundation.org/about/what-is-opc/>

Prieto, Albert. Industrial Shields. (2016) *Controlador PLC Industrial basado en Arduino, de Industrial Shield, Presentación modelo propio industria 4.0*. Recuperado de

<http://www.automaticaeinstrumentacion.com/es/downloads2/industrial-shields-albert-prieto.compressed.pdf>

Rockwell Automation. (s.f). *RS Logix 500 Solutions from Rockwell Software*, Recuperado de <http://www.rockwellautomation.com/rockwellsoftware/products/rslogix500.page>

Rockwell Automation. (s.f). *Sistemas de controlador logico programable Micrologix 1100*, Recuperado de <http://ab.rockwellautomation.com/es/Programmable-Controllers/MicroLogix-1100#overview>

Rockwell Automation. Supersedes Publication (2005). *ML 1100 series B Profile* 1763-PP001A-EN-P – July 2005 En Publication 1763-PP001B-EN-P – 2007— (Eds.) Disponible en https://literature.rockwellautomation.com/idc/groups/literature/documents/pp/1763-pp001_-en-p.pdf

SSIGSA. (s.f). *Automatización y control: Integrando y desarrollando tecnología*. Recuperado de <http://ssigsa.com/auto.php#>

Valencia H., Jose V. (Autor). (2018). *Diseño y aplicación de un sistema bajo arquitectura cliente servidor para la activación de riego automatizado en Arduino a traves de Twitter*, página 5 y 6, Recuperado de https://www.researchgate.net/publication/328174883_Disenio_y_aplicacion_de_un_sistema_bajo_arquitectura_cliente_servidor_para_la_activacion_de_riego_automatizado_en_Arduino_a_traves_de_Twitter